Routing and Wavelength Assignment with Protection: A Quadratic Unconstrained Binary Optimization Approach

Oylum Şeker[a,*], Merve Bodur[a], Hamed Pouya[a]

[a] *Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, ON, M5S3G8, Canada*

Abstract

The routing and wavelength assignment with protection is an important problem in telecommunications. Given an optical network and incoming connection requests, a commonly studied variant of the problem aims to grant maximum number of requests by assigning lightpaths at minimum network resource usage level, while ensuring the provided services remain functional in case of a single-link failure through dedicated path protection. We consider a practically relevant version where alternative lightpaths for requests are assumed to be given as a precomputed set, and show that it is NP-hard. We formulate the problem as an integer programming (IP) model, and also use it as a foundation to develop a novel quadratic unconstrained binary optimization (QUBO) model, which can be both directly solved by a state-of-the-art solver like GUROBI. We present necessary and sufficient conditions on objective function parameters to prioritize request granting objective over wavelength-link usage for both models, and a sufficient condition to ensure the exactness of the QUBO model. Moreover, we implement a problem-specific branch-and-cut algorithm for the IP model, and employ a new quantum-inspired technology, Digital Annealer (DA), for the QUBO model. We conduct computational experiments on a large suite of instances that are hard to optimally solve in order to assess the efficiency and efficacy of all of these approaches as well as a problem-specific heuristic. The results show that the emerging technology DA outperforms the considered established techniques coupled with GUROBI, in finding mostly significantly better or as good solutions in only two minutes compared to two hours of run time, whereas the problem-specific heuristic fails to be competitive.

Keywords: OR in telecommunications; routing and wavelength assignment; quadratic unconstrained binary optimization; Digital Annealer; integer programming

[*] Corresponding author

Email addresses: oylum.seker@utoronto.ca (Oylum Şeker), bodur@mie.utoronto.ca (Merve Bodur), h.pouya@utoronto.ca (Hamed Pouya)

June 4, 2021

1. Introduction

An optical network is a medium for information transmission via signals encoded in pulses of light. It connects devices that generate or store data through optical fibers carrying light channels. With the wavelength division multiplexing (WDM) technology allowing multiple signals to be simultaneously transmitted on the same fiber, optical networks have become particularly potent in conveying high volumes of information at very high speeds in a reliable way. As such, they are being increasingly deployed to meet the rapidly growing demand in many high-bandwidth applications such as real-time multimedia streaming, cloud computing and mobile network services (Majumdar, 2018; Chadha, 2019).

Wavelength-routed networks form a broad class of WDM networks, and can be considered as a set of nodes joined by fiber links. The communication between a pair of nodes is established through lightpaths, which is referred to as *connecting* these two nodes. A *lightpath* is an optical communication channel between two nodes in the network, and is comprised of a path (route) and a wavelength. A typical problem arising in wavelength-routed networks is the *routing and wavelength assignment* (RWA) problem. Given a set of connection requests between pairs of nodes, the RWA problem decides which requests to *grant*, i.e., provision a lightpath, such that no two allocated lightpaths with the same wavelength traverse a common fiber link, to prevent interference.

There are many variants of the RWA problem, which mainly differ in their objective and the nature of the connection demand process, as well as extensions incorporating additional concerns such as failures in the network, service quality and resource usage profile (Bandyopadhyay, 2007). In general, they are all difficult to solve due to the inherent computational complexity of the RWA problem (Erlebach and Jansen, 2001).

In this paper, we study an RWA problem whose features are motivated by practical concerns (e.g., fast recovery from failures), and some common goals in telecommunications industry (such as granted request maximization and resource usage minimization). In the sequel, we first provide all the relevant background information and motivate our problem setup (Section 1.1); then formally define our problem, introduce our solution approach and summarize our contributions (Section 1.2).

1.1. Background Information

The information provided in this section is mostly based on the books by Mukherjee (2006), Bandyopadhyay (2007) and Chatterjee et al. (2016).

Network failures and recovery schemes.

While it is desirable to maximize the number of granted requests in RWA, it is also often necessary to provide some degree of protection for them against potential failures in the network. In a WDM network, any of the components may fail, and one failure may disrupt multiple connections. Link failure is the most frequently encountered type of fault, which may for instance arise from fiber cuts due to human errors during construction operations, or natural calamity such as earthquakes. Also, the probability of having multiple link failures simultaneously, or having additional failure(s) before one has been repaired, is considered to be negligible. Thus, the literature is mostly concerned with the case of a *single-link failure*, which we follow as well.

As each fiber link can carry terabits of data per second, even a brief disruption of a connection can result in a large amount of data loss. As such, fault management mechanisms play an important role in WDM network survivability (Zhang and Mukheriee, 2004). Link failures can be handled at the optical or a higher layer, but the time to detect and repair a failed link ranges from tens of seconds to a few days. Since even the shortest repair time is still too long relative to the rate of data transfer, some *fault recovery strategies*, which re-route the broken connections using the available network resources, are usually adopted to keep the services functional while the repair process is in progress. There are two main categories of such strategies, *protection* and *restoration*. In the protection scheme, *backup (protection) lightpaths* are computed and reserved in advance along with the *primary (working) lightpaths*. This ensures fast recovery of all the affected connections by replacing the primary lightpaths with the backups (in milliseconds upon failure), at the expense of increased network resource usage. In the restoration scheme, on the other hand, backup capacity is not provisioned prior to the occurrence of failure, instead new lightpaths are discovered dynamically upon the interruption of connections. Despite being less demanding in resource usage, restoration schemes fail to guarantee resource availability, and lead to higher recovery times.

Protection and restoration schemes for link failures can be categorized with respect to being *path-* or *link*-based, i.e., whether they re-route the whole path or only the failed link(s). Path protection schemes can be *dedicated* or *shared*. When it is dedicated, each backup lightpath is reserved for only one primary lightpath, and two backup lightpaths with the same wavelength cannot have a common link on their routes. Two commonly used subcategories of this scheme are denoted by 1:1 and 1+1. The former transmits data through only primary lightpaths before failure, while the latter allows the use of both primary and backup lightpaths simultaneously.

Static and dynamic RWA.

The underlying demand process in RWA problems is considered as *static* or *dynamic*. The static case assumes that connection requests with their associated source and destination nodes are known in advance. In the dynamic case, on the other hand, requests arrive one by one and are provisioned a lightpath in real-time. In our study, we aim to efficiently solve the static RWA problem.

We note that a dynamic RWA problem can be approximated by discretizing the time horizon into intervals and solving a static RWA problem for each using the batch of requests arrived during the corresponding interval, as typically done for dynamic network reconfiguration problems (Zhang et al., 2007; Wu et al., 2012; Grover, 2013). Such an approach would usually require solving the static problems quickly, e.g., for some bandwidth on demand services, their service level agreement between the provider and the customer guarantees connections to be established (with a certain level of protection) in a matter of minutes (Fawaz et al., 2004; Losego et al., 2005). In that regard, the methods solving the static RWA problem efficiently may well benefit dynamic settings. In fact, one method we propose can generate good-quality solutions in up to two minutes.

Precomputed paths.

One way to improve solution times for RWA problems is to employ a two-phase framework; generating a set of paths between all or some potential source and destination pairs in the first phase, and picking one from the precomputed alternatives for each request and doing the wavelength assignment in the second phase, for instance as adopted in (Li and Simha, 2000; Noronha and Ribeiro, 2006). In cases where an RWA problem needs to be solved repeatedly over time, this strategy can be made more efficient by performing the first phase only once at the beginning, i.e., by using the same set of precomputed paths throughout the horizon. Granting requests from their precomputed set of alternatives may also provide more control to decision makers, in the sense that they can disperse the demand over the network in a balanced way if desired, and can have a better idea on the state of the network for subsequent decision-making stages well in advance.

1.2. Our Work

Problem definition.

In the light of discussions in the background section, in this study, we consider the RWA problem with static demand and 1:1 dedicated path protection scheme against single-link failures, for a given network with a set of precomputed alternative primary and backup paths and a number of available wavelengths. We call this the *Routing and Wavelength Assignment with Protection*

(RWA-P) problem, where the aim is to primarily maximize the number of granted requests, since this brings the actual gains for the service providers (Shen et al., 2005), while minimizing the wavelength-link usage as a secondary goal to save network resources for future demands.

Outline of our work.

We show that RWA-P is NP-hard, propose mathematical models and evaluate their solution performance using promising technologies. More specifically, we propose an integer programming (IP) model for RWA-P as well as a strengthened version of it, and use it as a foundation to develop a novel quadratic unconstrained binary optimization (QUBO) model. For model parameters, we present conditions to achieve the intended objective prioritization of granted requests over resource usage, as well as to ensure that the QUBO model is exact. We use the exact state-of-the-art solver GUROBI to solve the IP models, both by directly providing them to the solver and also applying a problem-specific branch-and-cut method. In order to derive solutions for the QUBO model, we employ a new technology called Digital Annealer (DA), and also test GUROBI. We conduct computational experiments on a large suite of instances based on commonly used networks, compare the efficiency and efficacy of all our methods as well as a problem-specific heuristic from the literature, and analyze the impact of the penalty coefficient parameter of the QUBO model on DA's performance.

Choice of solution technologies.

IP has been a widely adopted modeling framework for discrete optimization problems, including RWA problems. This can be attributed to the development of many efficient solution methodologies and enhancements such as the branch-and-cut algorithm, presolve techniques and heuristics. Despite their advanced status, state-of-the-art IP solvers may take prohibitively long times to yield optimal solutions, especially for large-scale problems, because run times are typically exponential in the size of the input model. However, for many practical problems, it is sufficient to obtain a good-quality solution, as also noted in (Giovanni, 2017), but preferably in a reasonable/short amount of time. Although IP solvers can be tuned to generate feasible solutions faster, they may fail to address both of these practical considerations simultaneously.

A plausible alternative to tackle such problems is to formulate them as QUBO models and generate solutions via novel computational architectures and new technologies, such as adiabatic quantum computing (e.g., (Papalitsas et al., 2019)), neuromorphic computing (e.g., (Corder et al., 2018)), and optical parametric oscillators (e.g., Inagaki et al. (2016)), which have recently attracted significant attention due to their capability in tackling combinatorial optimization problems. A promising example to these new technologies is DA (Aramon et al., 2019), which is a computer architecture that rivals quantum computers in utility (Boyd, 2018). DA is designed to solve QUBO models, and uses an algorithm based on simulated annealing. In many applications, such as minimum vertex cover problem (Javad-Kalbasi et al., 2019), maximum clique problem (Naghsh et al., 2019) and outlier rejection (Rahman et al., 2019), it has been shown to significantly improve upon the state of the art and yield high-quality solutions in radically short amount of time.

Contributions.

The contributions of this paper are as follows:

- We prove that RWA-P is NP-hard.
- We develop IP and QUBO models, and propose conditions on parameter values to ensure their validity in the sense that the intended objective prioritization is achieved. Moreover, we propose sufficient conditions establishing the exactness of the QUBO model.
- Our QUBO approach is the first of its kind in the vast RWA literature and observed to be highly efficient and efficacious in solving a large suite of nontrivial test instances, well addressing the practical needs of RWA-P, which signifies that it can potentially be useful for other important RWA problems due to structural similarities.
- We show that the emerging DA technology outperforms highly established IP methods accompanied by advanced solvers in handling instances that are hard to optimally solve, indicating that it has the potential to become a viable tool in addressing combinatorial optimization problems. Considering that DA is rarely employed in the operations research literature and only few studies compare it with the state of the art, e.g., (Şeker et al., 2020; Naghsh et al., 2019; Ohzeki et al., 2019; Matsubara et al., 2020), our study serves as a step to bridge the gap between the use of the established and new promising solution technologies.
- We show through sensitivity analysis that smaller penalty coefficient values lead to better solutions for DA, which has been mentioned in only few studies previously, e.g., (Cohen et al., 2020a,b). To the best of our knowledge, this is the first reported systematic analysis of the penalty coefficient on DA solution quality.

Paper outline.

The remainder of this paper is organized as follows. We review the related literature in Section 2. In Section 3, we present an IP formulation for RWA-P, provide a prioritization condition and discuss problem complexity. Then, we introduce a QUBO model for RWA-P in Section 4, derive a condition that render it exact, and overview the operating principles of DA. In Section 5, we present our computational study, and finally in Section 6, we conclude our paper with a brief summary.

2. Related Literature

There are many variants of the RWA problem and a vast number of studies on each type. Here, we restrict our review to the RWA problem with dedicated path protection scheme and static demand, and summarize the most relevant studies in the sequel.

Ramamurthy et al. (2003) examine different protection approaches for single-link failures, and develop IP formulations for path- and link-based schemes, assuming that a precomputed set of alternate routes are given. Their model for dedicated path protection aims to minimize the total number of wavelengths used over all links in the network, while enforcing that all the demand is satisfied and the wavelength capacity on the links is not exceeded. Using test instances generated for a representative network topology, they compare the performance of the proposed IP models. Their problem setup differs from ours in that they do not allow unsatisfied demand assuming sufficient capacity in the network, as such use a different objective than we do.

Azodolmolky et al. (2010) study the RWA problem with dedicated path protection, where they assume a precomputed set of pairs of primary and backup paths per request, i.e., each primary path has an associated unique backup path. They present two IP models, which form the basis of their heuristic algorithm designed for the impairments aware RWA problem. The first model considers the requests that require only a primary lightpath, and aims to minimize the total number of requests that are not accepted. The second model extends the first one by (i) adding another demand category that necessitates backup lightpaths as well, and (ii) minimizing a combined sum of the former objective and the maximum number of times a wavelength is used on a link, in a way that acceptance of the requests requiring protection are prioritized over the ones that do not, and the wavelength usage is of least priority. The problem setting used for their IP models is similar to ours, especially when it is assumed that no unprotected demand exists. However, it does not allow the primary and backup path options of a request to be arbitrarily combined; they are instead considered in predefined pairs only. The authors test the performances of their proposed heuristic algorithm, two versions of the heuristic from the work of Ezzahdi et al. (2006) that they enhance with quality of transmission considerations, and solving of the presented IP formulations via an exact solver. In our study, we also test the heuristic by Ezzahdi et al. (2006) for RWA-P and compare it to our presented methods in Section 5.

We note that, in addition to the aforementioned problem setup differences, our study stands apart from those works in terms of modelling. While the previous IP formulations are link-based, our models are path-based. Also, we present an exact QUBO model, the first for an RWA problem although a *constrained* binary quadratic model has been used in (Ebrahimzadeh et al., 2013).

There are some other relevant works that consider the RWA problem with protection for single-link (or node) failures, either with precomputed paths (Lee and Park, 2006), or by solving both the routing and the wavelength assignment problems simultaneously (Wang et al., 2001; Li Shifeng et al., 2002) or sequentially (Kokkinos et al., 2010).

Lastly, we note that for many variants of the RWA problem, the complexity has been established to be NP-hard, see for instance (Erlebach and Jansen, 2001; Li and Simha, 2000; Chlamtac et al., 1992; Chiu and Modiano, 2000). Despite some structural similarities between those variants and our problem, the complexity of RWA-P has remained open.

3. IP Formulation and Problem Complexity

In this section, we first present an IP formulation for the RWA-P problem in Section 3.1, followed by Section 3.2 where we propose values for the weight parameters used in combining two objectives, namely granted request maximization and link usage minimization, into a single one in order to provably achieve the desired prioritization of the former over the latter. We then show the complexity of RWA-P in Section 3.4 through a reduction from a well-known NP-complete problem.

3.1. IP Formulation

As formally defined in Section 1.2, the RWA-P problem aims to grant maximum number of requests by properly assigning a working and a protection lightpath to each from a precomputed collection, while minimizing the wavelength-link usage as a secondary goal, which we hereafter refer to as link usage for simplicity.

We model an optical network as a directed graph $G = (V, E)$, with V and E respectively denoting the set of nodes and the set of directed edges that join ordered pairs of nodes, where it is possible to have multiple arcs with the same start and end nodes. In telecommunications context, we refer the directed edges of the input graph as links. We denote the set of requests by R, and the set of wavelengths by Λ. For each request $r \in R$, we represent the set of alternative working and protection lightpaths with W^r and P^r, respectively, which are obtained by combining the available precomputed set of paths and wavelengths. The length of a working (protection) lightpath w (p) for request $r \in R$, i.e., the number of the links it contains, is denoted by B^r_w (B^r_p). For convenience, we use $E[\ell]$ to represent the set of links that a given lightpath ℓ contains, and $\Lambda[\ell]$ for the wavelength associated with ℓ.

In order to help compactly represent the constraints of RWA-P, we define four *conflict sets*, $\mathcal{C}_1, \ldots, \mathcal{C}_4$. The first conflict set $\mathcal{C}_1$ serves to enforce the pair of working and protection lightpaths for a given request to be link-disjoint. It is comprised of (r, w, p) triplets such that the working lightpath w and the protection lightpath p for request r have at least one link in common. Namely,

$$\mathcal{C}_1 \coloneqq \{(r, w, p)\colon r \in R,\ w \in W^r,\ p \in P^r,\ E[w] \cap E[p] \neq \varnothing\}.$$

The remaining three conflict sets are used to prevent the concurrent use of lightpaths having the same wavelength and sharing a link. Considering such lightpaths in pairs, there can be one working and one protection, two working, or two protection lightpaths fulfilling these criteria, which we address through sets $\mathcal{C}_2$, $\mathcal{C}_3$, and $\mathcal{C}_4$, respectively. Let $\mathcal{C}_2$ be the set of (r_1, r_2, w, p) quadruplets such that the working and protection lightpaths w and p for distinct requests r_1 and r_2 have the same wavelength and at least one link in common:

$$\mathcal{C}_2 \coloneqq \{(r_1, r_2, w, p)\colon r_1, r_2 \in R,\ r_1 \neq r_2,\ w \in W^{r_1},\ p \in P^{r_2},\ \Lambda[w] = \Lambda[p],\ E[w] \cap E[p] \neq \varnothing\}.$$

The sets $\mathcal{C}_3$ and $\mathcal{C}_4$ contain a similar collection of quadruplets as $\mathcal{C}_2$ does, but with only working and only protection lightpaths, respectively:

$$\mathcal{C}_3 \coloneqq \{(r_1, r_2, w_1, w_2)\colon r_1, r_2 \in R,\ w_1 \in W^{r_1},\ w_2 \in W^{r_2},\ \Lambda[w_1] = \Lambda[w_2],\ E[w_1] \cap E[w_2] \neq \varnothing\},$$

$$\mathcal{C}_4 \coloneqq \{(r_1, r_2, p_1, p_2)\colon r_1, r_2 \in R,\ p_1 \in P^{r_1},\ p_2 \in P^{r_2},\ \Lambda[p_1] = \Lambda[p_2],\ E[p_1] \cap E[p_2] \neq \varnothing\}.$$

Example 1. In Figure 1, an example RWA-P network is illustrated with two requests together with their working and protection lightpath alternatives. The source and destination nodes of the two requests are those with s^r and t^r labels for $r \in \{1, 2\}$, respectively. The lightpaths are shown with red and green, where each color symbolizes a distinct wavelength, and the lines being solid or dashed indicate whether the lightpath is in the working or protection set, respectively. The request and working/protection indices of the lightpaths are shown beside them in the same color as the lines representing them. For request 1, there are three working and one protection lightpaths, and for request 2, there is one working and two protection lightpaths.

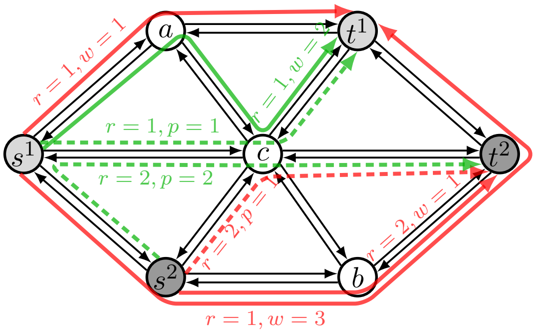

Figure 1: An RWA-P network and two requests with their precomputed working and protection lightpaths.

Let us give some example tuples for the conflict sets using the network in Figure 1. The link (c, t^1) is common in the lightpaths labeled with $r = 1, w = 2$ and $r = 1, p = 1$, which yields $(r, w, p) = (1, 2, 1) \in \mathcal{C}_1$. Furthermore, the link (s^2, b) being contained in two lightpaths having the same wavelength (red) makes $(r_1, r_2, w_1, w_2) = (1, 2, 3, 1) \in \mathcal{C}_3$, and (s^1, c) being shared by two protection lightpaths with the same wavelength (green) leads to $(r_1, r_2, p_1, p_2) = (1, 2, 1, 2) \in \mathcal{C}_4$. Since there is no pair of distinct requests whose working and protection lightpaths have the same wavelength, $\mathcal{C}_2 = \varnothing$ here.

In this example, it is possible to accept both of the requests by selecting the working and protection lightpaths $w = 1$ and $p = 1$ for request $r = 1$, and also for $r = 2$. This solution is indeed the best option for link usage as well, because having granted all of the given requests with lightpaths of length two, it is not possible to use any fewer links as each lightpath is of length at least two in this example.

Using the notation introduced above and two sets of binary decision variables defined as

$$x_w^r = \begin{cases} 1, & \text{if working lightpath } w \in W^r \text{ is assigned to request } r \in R \\ 0, & \text{otherwise} \end{cases}$$

$$y_p^r = \begin{cases} 1, & \text{if protection lightpath } p \in P^r \text{ is assigned to request } r \in R \\ 0, & \text{otherwise} \end{cases}$$

we now present a novel IP formulation as follows:

$$
\begin{aligned}
\min \quad & f(x,y) \coloneqq \alpha \sum_{r\in R}\left(\sum_{w\in W^r} B_w^r x_w^r + \sum_{p\in P^r} B_p^r y_p^r\right) - \beta \sum_{r\in R}\sum_{w\in W^r} x_w^r && && \text{(1a)}\\
\text{s.t.} \quad & \sum_{w\in W^r} x_w^r - \sum_{p\in P^r} y_p^r = 0 && r\in R && \text{(1b)}\\
& \sum_{w\in W^r} x_w^r \le 1 && r\in R && \text{(1c)}\\
& x_w^r + y_p^r \le 1 && (r,w,p)\in \mathcal{C}_1 && \text{(1d)}\\
& x_w^{r_1} + y_p^{r_2} \le 1 && (r_1,r_2,w,p)\in \mathcal{C}_2 && \text{(1e)}\\
& x_{w_1}^{r_1} + x_{w_2}^{r_2} \le 1 && (r_1,r_2,w_1,w_2)\in \mathcal{C}_3 && \text{(1f)}\\
& y_{p_1}^{r_1} + y_{p_2}^{r_2} \le 1 && (r_1,r_2,p_1,p_2)\in \mathcal{C}_4 && \text{(1g)}\\
& x_w^r,\ y_p^r \in \{0,1\} && r\in R,\ w\in W^r,\ p\in P^r && \text{(1h)}
\end{aligned}
$$

where α and β are predetermined positive constants.

Constraint set (1b) enforces that the same number of working and protection lightpaths are selected to grant a request, and (1c) ensures that at most one working lightpath is assigned to each request. Constraint set (1d) guarantees that the selected working and protection lightpaths for each request are link-disjoint, while (1e)–(1g) make sure that the lightpaths having the same wavelength and sharing a link are not chosen simultaneously. Finally, constraint set (1h) states the domains of the decision variables.

The objective function (1a) combines the two goals of RWA-P, minimizing the number of links used and maximizing the number of requests granted, as a weighted sum. As mentioned in the introduction, the latter goal must be prioritized over the former, which we detail next.

3.2. Objective Prioritization

We now formally define what prioritization of request granting over link usage means, and propose α and β values that serve the purpose in (1a). We first introduce some notation to be used in the sequel. Let $f_\alpha(x,y)$ and $f_\beta(x)$ be two functions respectively corresponding to the number of links used and the number of requests granted at a solution, i.e.,

$$
f_\alpha(x,y) \coloneqq \sum_{r\in R}\left(\sum_{w\in W^r} B_w^r\, x_w^r + \sum_{p\in P^r} B_p^r\, y_p^r\right), \quad f_\beta(x) \coloneqq \sum_{r\in R}\sum_{w\in W^r} x_w^r,
$$

so that the objective function (1a) can be equivalently written as

$$f(x, y) \; = \; \alpha \; f_\alpha(x, y) \; - \; \beta \; f_\beta(x).$$

Furthermore, to ease the presentation, for any given feasible solution $(\overset{\bullet}{x}, \overset{\bullet}{y})$ (where $\bullet$ represents any operator such as hat, tilde, and bar), we respectively define the associated IP objective value and its components as

$$\overset{\bullet}{f} \; \coloneqq \; f(\overset{\bullet}{x}, \overset{\bullet}{y}), \quad \overset{\bullet}{f}_\alpha \; \coloneqq \; f_\alpha(\overset{\bullet}{x}, \overset{\bullet}{y}), \quad \overset{\bullet}{f}_\beta \; \coloneqq \; f_\beta(\overset{\bullet}{x}),$$

and the worst-case and best-case link usage of a feasible solution granting the same number of requests as

$$\begin{aligned} \overset{\bullet}{f}{}_\alpha^{\max} \; &\coloneqq \max \left\{ f_\alpha(x, y) \mid (x, y) \text{ satisfies } (1b) - (1h), f_\beta(x) = \overset{\bullet}{f}_\beta \right\}, \\ \overset{\bullet}{f}{}_\alpha^{\min} \; &\coloneqq \min \left\{ f_\alpha(x, y) \mid (x, y) \text{ satisfies } (1b) - (1h), f_\beta(x) = \overset{\bullet}{f}_\beta \right\}. \end{aligned}$$

Definition 1 (Prioritization Condition)**.** *Request granting is prioritized over link usage, if for any pair of feasible solutions $(\hat{x}, \hat{y})$ and $(\tilde{x}, \tilde{y})$ with $\hat{f}_\beta > \tilde{f}_\beta$, we have $\hat{f} < \tilde{f}$, i.e., the marginal contribution of granting a request to the objective function (1a) is always negative. That is, $\alpha \hat{f}_\alpha - \beta \hat{f}_\beta < \alpha \tilde{f}_\alpha - \beta \tilde{f}_\beta$ for all feasible $(\hat{x}, \hat{y})$ and $(\tilde{x}, \tilde{y})$ with $\hat{f}_\beta > \tilde{f}_\beta$.*

Considering the largest and smallest realizations of the left- and right-hand sides in terms of link usage, respectively, the prioritization condition can be equivalently written as

$$\alpha \hat{f}_\alpha^{\max} - \beta \hat{f}_\beta \; < \; \alpha \tilde{f}_\alpha^{\min} - \beta \tilde{f}_\beta \;\; \text{for all feasible } (\hat{x}, \hat{y}) \text{ and } (\tilde{x}, \tilde{y}) \text{ with } \hat{f}_\beta > \tilde{f}_\beta. \tag{2}$$

This condition can also be expressed with the help of an optimization model:

$$\frac{\beta}{\alpha} \; > \; \Omega^{>} \; \coloneqq \; \max \left\{ \frac{\hat{f}_\alpha^{\max} - \tilde{f}_\alpha^{\min}}{\hat{f}_\beta - \tilde{f}_\beta} : (\hat{x}, \hat{y}) \text{ and } (\tilde{x}, \tilde{y}) \text{ are feasible with } \hat{f}_\beta > \tilde{f}_\beta \right\}. \tag{3}$$

Indeed, it is possible define another optimization model by only considering the solutions differing by one in their number of granted requests,

$$\Omega^{=} \; \coloneqq \; \max \left\{ \frac{\hat{f}_\alpha^{\max} - \tilde{f}_\alpha^{\min}}{\hat{f}_\beta - \tilde{f}_\beta} : (\hat{x}, \hat{y}) \text{ and } (\tilde{x}, \tilde{y}) \text{ are feasible with } \hat{f}_\beta = \tilde{f}_\beta + 1 \right\}, \tag{4}$$

which would achieve what (3) does, as provided in the proposition below.

Proposition 1. *The optimization models in (3) and (4) yield the same optimal values; that is, $\Omega^{>} = \Omega^{=}$. This implies that the prioritization condition given in (3) can also be achieved by setting $\alpha, \beta > 0$ such that $\frac{\beta}{\alpha} > \Omega^{=}$.*

Proof. See Appendix A.1. □

For practical purposes, we assume $\alpha = 1$ and that β can only take integer values. In this case, letting β^{Tight} denote the smallest integer β value satisfying the prioritization condition in (4), we have $\beta^{\text{Tight}} = 1 + \Omega^{=}$. However, obtaining β^{Tight} or at least a reasonable upper bound for it necessitates solving of a non-trivial optimization problem, which could be computationally even more expensive than the original RWA-P problem. Next, we derive sufficient condition for prioritization in terms of given instance parameters.

Proposition 2 (IP objective weight selection)**.** *Selecting $\alpha, \beta > 0$ such that*

$$\frac{\beta}{\alpha} \ > \ |R| \ (M-2) \ + \ 2 \tag{5}$$

prioritizes request granting over link usage in (1a) for any feasible solution to the IP, i.e., solutions accepting more requests yield lower objective values, where $M = \max_{r\in R} \left\{ \max_{w\in W^r} \left\{B^r_w\right\} + \max_{p\in P^r} \left\{B^r_p\right\} \right\}$.

Proof. See Appendix A.2. □

Proposition 2 provides a lower bound on $\frac{\beta}{\alpha}$ that is sufficient to make request granting the primary goal. Note that computing this lower bound does not involve solution of an optimization problem and hence makes it easy to decide on safe objective parameter combinations for a given instance. Letting β^{Base} denote the least possible β value from the condition in (5) (assuming $\alpha = 1$), we have $\beta^{\text{Base}} \ = \ |R| \ (M-2) \ + \ 3$. Note that by definition $\beta^{\text{Base}} \geq \beta^{\text{Tight}}$.

We now show that there exist examples where the bound in (5) is indeed tight.

Proposition 3 (Tight example for the weight selection)**.** *There exist RWA-P instances for which the lower bound provided in Proposition 2 is necessary to prioritize request granting over link usage.*

Proof. See Appendix A.3. □

3.3. Strengthened Conflict Constraints

We can strengthen the set of conflict constraints in (1d)–(1g) by identifying larger groups of variables that are mutually in conflict, i.e., groups in which at most one variable can take value one. To this end, we first construct the set $\bar{P}^{r,w}$ of all protection lightpaths for request r that has at least one common link with working lightpath w, as an extension of the lightpath pairs defined in $\mathcal{C}_1$. That is, for every $w \in W^r$ and request $r \in R$, we define

$$\bar{P}^{r,w} := \left\{p \in P^r \colon E[w] \cap E[p] \neq \varnothing\right\}.$$

Next, we define the sets of working and protection lightpaths that contain link e and wavelength λ, for every wavelength $\lambda \in \Lambda$, link $e \in E$, and request $r \in R$, which will extend the lightpath pairs defined in conflict sets $\mathcal{C}_2, \mathcal{C}_3$ and $\mathcal{C}_4$. Namely,

$$\tilde{W}^r_{e,\lambda} \coloneqq \big\{ w \in W^r \colon \Lambda[w] = \lambda,\ e \in E[w] \big\},$$

and

$$\tilde{P}^r_{e,\lambda} \coloneqq \big\{ p \in P^r \colon \Lambda[p] = \lambda,\ e \in E[p] \big\}.$$

Using these sets, we can write a strengthened form of our conflict constraints as

$$x^r_w \ + \sum_{p \in \bar{P}^{r,w}} y^r_p \ \le \ 1 \qquad\qquad r \in R,\ w \in W \tag{6a}$$

$$\sum_{r \in R} \sum_{w \in \tilde{W}^r_{e,\lambda}} x^r_w \ + \sum_{r \in R} \sum_{p \in \tilde{P}^r_{e,\lambda}} y^r_p \ \le \ 1 \qquad\qquad e \in E,\ \lambda \in \Lambda. \tag{6b}$$

In the rest of the paper, we refer to the IP model in (1) as IP^{Base}, and to that with the conflict constraints in (1d)–(1g) being replaced with the ones in (6a)–(6b) as $\text{IP}^{\text{Strong}}$.

3.4. Complexity

In this section, we prove that the RWA-P problem is NP-hard. More specifically, we prove that a special case of our problem is NP-hard by making a reduction from the *maximum stable set* problem, which is known to be NP-hard (Garey and Johnson, 1979).

Given a graph $G_s\,(V_s, E_s)$, a *stable set* is a set $V'_s \subseteq V_s$ of nodes such that no two nodes in V'_s are linked by an edge in E_s. The goal in the maximum stable set problem (MSS) is to find a stable set of maximum cardinality in G_s. The decision version of the problem, which we denote by Dec-MSS, is concerned with the existence of a stable set of size at least k in the input graph G_s, for some given integer $k \ge 1$.

Now, we define RWA-P-r as the special version of the RWA-P problem that aims to maximize the number of granted requests only (thus the extension "-r"), i.e., the variant which can be modeled as (1) using $\alpha = 0$ and $\beta = 1$ in the objective function (1a). Let Dec-RWA-P-r be the decision version of RWA-P-r, which checks whether at least k requests can be granted for some given integer $k \ge 1$. In what follows, we show that RWA-P-r is NP-hard, starting with the polynomial-time verifiability for its decision version.

Lemma 1 (Verifiability)**.** Dec-*RWA-P*-r *is in NP.*

Proof. See Appendix A.4. □

Theorem 1 (Complexity)**.** *RWA-P-*R *is NP-hard.*

Proof. See Appendix A.5. □

As RWA-P generalizes RWA-P-R, it is at least as hard, which yields the desired result.

Corollary 1. *RWA-P is NP-hard.*

4. QUBO Formulation and Solution Method

In this section, we present our proposed modeling and solution approach for the RWA-P problem. In Section 4.1, we present a QUBO model via a transformation from IP^{Base} obtained by dualizing its constraints, i.e., adding them as a penalty term to the objective function. We also explain how to carefully choose the penalty parameters to achieve the exactness of the QUBO model. Then, in Section 4.2, we overview the Digital Annealer technology and its operating principles.

4.1. Transformation to QUBO

As the first step of obtaining an exact QUBO formulation, we dualize the constraints of our IP formulation given in (1), IP^{Base}, in such a way that any infeasible solution to it, i.e., any constraint violation, yields a strictly positive penalty term in the objective function of our QUBO model. This is easy to achieve for equality constraints; any linear equality constraint can be transformed into a penalty term by simply taking the square of the difference of its left and right-hand sides, so that any constraint violation translates into a positive penalty value, and hence can be avoided through the minimization of the objective. Therefore, for our only set of equality constraints (1b), the corresponding penalty term includes for each $r \in R$ the following squared violation expression:

$$\left(\sum_{w \in W^r} x_w^r - \sum_{p \in P^r} y_p^r \right)^2 ,$$

which amounts to a positive value when more working lightpaths than protection lightpaths are selected for the request r, or vice versa.

In case of inequality constraints, however, more custom-tailored approaches are needed, because violations occur in one direction only. In order to transform the inequality constraints in (1c)–(1g) into penalty terms, we first reformulate them as quadratic *equality* constraints in (7).

$$\left(\sum_{w \in W^r} x_w^r\right)\left(\sum_{w \in W^r} x_w^r - 1\right) = 0 \qquad r \in R \tag{7a}$$

$$x_w^r \; y_p^r = 0 \qquad (r, w, p) \in \mathcal{C}_1 \tag{7b}$$

$$x_w^{r_1} \; y_p^{r_2} = 0 \qquad (r_1, r_2, w, p) \in \mathcal{C}_2 \tag{7c}$$

$$x_{w_1}^{r_1} \; x_{w_2}^{r_2} = 0 \qquad (r_1, r_2, w_1, w_2) \in \mathcal{C}_3 \tag{7d}$$

$$y_{p_1}^{r_1} \; y_{p_2}^{r_2} = 0 \qquad (r_1, r_2, p_1, p_2) \in \mathcal{C}_4 \tag{7e}$$

Constraint set (7a) is the quadratic equivalent of (1c) ensuring that at most one lightpath is selected per request. As the decision variables are binary, the left-hand side of (1c), i.e., the expression denoting the total number of working lightpaths assigned to request r, can take value either zero or one, in which case the left-hand side of (7a) becomes zero. So, (7a) holds only when the corresponding original constraint (1c) is satisfied, otherwise, i.e., when $\sum_{w \in W^r} x_w^r \geq 2$, the left-hand side of (7a) takes a strictly positive value. Therefore, the left-hand side of (7a) can be used as a penalty term for violations of constraints (1c). Similarly, constraints (1d)–(1g), each of which ensuring that the two involved lightpaths cannot be both selected due to a conflict, are violated only when both variables on the left-hand side take value one; all other configurations of the two binary variables are feasible. The quadratic constraints (7b)–(7e) take advantage of the fact that all feasible configurations involve at least one variable having value zero, and force the product of the two to be zero. So, when the associated constraints are violated, the left-hand sides of (7b)–(7e) take strictly positive values, namely value one, thus serve as penalty terms to be added to the objective function of our QUBO model. Note that the magnitude of violation that an infeasible binary solution creates in any one of the constraints (1b)–(1g) is at least one, which has a useful role in rendering our QUBO formulation exact, as we will see when we specify possible values of the penalty coefficient in the sequel.

We present our QUBO formulation for RWA-P in (8).

$$\begin{aligned}
\min \quad & \alpha \sum_{r\in R}\left(\sum_{w\in W^r} B^r_w x^r_w + \sum_{p\in P^r} B^r_p y^r_p\right) - \beta \left(\sum_{r\in R}\sum_{w\in W^r} x^r_w\right) \\
& + \rho \sum_{r\in R}\left(\sum_{w\in W^r} x^r_w - \sum_{p\in P^r} y^r_p\right)^2 + \rho \sum_{r\in R}\left(\sum_{w\in W^r} x^r_w - 1\right)\left(\sum_{w\in W^r} x^r_w\right) \\
& + \rho \sum_{(r,w,p)\in\mathcal{C}_1} x^r_w\ y^r_p \quad + \quad \rho \sum_{(r_1,r_2,w,p)\in\mathcal{C}_2} x^{r_1}_w\ y^{r_2}_p \\
& + \rho \sum_{(r_1,r_2,w_1,w_2)\in\mathcal{C}_3} x^{r_1}_{w_1}\ x^{r_2}_{w_2} \quad + \quad \rho \sum_{(r_1,r_2,p_1,p_2)\in\mathcal{C}_4} y^{r_1}_{p_1}\ y^{r_2}_{p_2} && \text{(8a)} \\
\text{s.t.} \quad & x^r_w,\ y^r_p \in \{0,1\} \qquad\qquad r\in R,\ w\in W^r,\ p\in P^r, && \text{(8b)}
\end{aligned}$$

where $\rho > 0$ is the penalty coefficient for the dualized constraints. We note that different penalty coefficients can be used for different terms, however, we choose them to be all the same, ρ, to simplify our derivation of a valid lower bound for it.

We note that IP$^{\text{Strong}}$ can also be transformed into a QUBO model, in which case the strengthened set of conflict constraints in (6a)–(6b) would simply be dualized in the same manner constraints in (1c) are dualized (see quadratic equalities in (7a)). This yields sums of bilinear penalty terms in the objective, similar to those corresponding to the conflict constraints in (1d)–(1g). Thus, the two QUBO models can be made equivalent by customizing the penalty coefficient values for the associated terms.

Next, we investigate conditions ensuring that an optimal solution to the QUBO model is also optimal for the RWA-P problem.

Definition 2 (Exactness)**.** *A model $\mathcal{M}$ for a problem $\mathcal{P}$ is* exact *if any optimal solution to $\mathcal{M}$ is feasible and optimal for $\mathcal{P}$.*

By construction, IP$^{\text{Base}}$ (and also IP$^{\text{Strong}}$) is an exact model for the RWA-P problem. On the other hand, for the QUBO formulation to be exact, the penalty coefficient ρ should be selected “sufficiently large”. Since high valued parameters might lead to serious numerical issues, smaller “safe” values are desirable. In that regard, we provide a lower bound for ρ that is sufficient to guarantee that the QUBO model is exact.

Proposition 4 (QUBO penalty selection)**.** *When*

$$\rho > \beta(|R| + 1) - \alpha \left(1 + \sum_{r \in R} (B^r_{w_{\min}} + B^r_{p_{\min}})\right), \tag{9}$$

(8) *is an exact QUBO model for the RWA-P problem, where* $B^r_{w_{\min}} = \min_{w \in W^r} \{B^r_w\}$ *and* $B^r_{p_{\min}} = \min_{p \in P^r} \{B^r_p\}$.

Proof. See Appendix A.6. □

It is important to note that the condition in (9) not only guarantees the exactness of the QUBO model, but also ensures that any infeasible solution for the problem is inferior to the feasible ones. We chose to impose this stronger requirement in deriving the lower bound on the penalty coefficient in order to establish a clear dominance relationship between the classes of feasible and infeasible solutions, which we believe leads to a conceptually better QUBO model. We also note that the resulting lower bound is not tight, as far as the original definition of exactness is concerned. If ρ^{Base} is a penalty coefficient abiding (9), then similar to the objective weight parameter discussion in the IP case, we can actually design an optimization model to obtain the smallest possible penalty coefficient value, ρ^{Tight}. However, the resulting model would be much more complex (e.g., a 0-1 quadratic fractional programming model).

Given a QUBO model, we can optimally solve it using the state-of-the-art solvers like GUROBI (Gurobi Optimization LLC, 2020) and CPLEX (IBM, 2019). However, if the model is originally constrained and linear, as it is in our case, a more favorable approach would be to use these solvers to solve the IP formulations, in which they are particularly successful. The main limitation of the IP solvers is that their performance deteriorates as the number of variables and constraints increases, with an exponential rise in solution times typically, as a result of which they likely fail to deliver an optimal or good-quality solution for realistic problem sizes in a short amount of time. For problems suitable to be formulated as a QUBO model, a promising alternative to the state of the art mentioned above is the *Digital Annealer* technology, which in theory is not affected by the increasing number of variables and constraints, and demonstrates a robust level of performance across instances having different sizes, as long as the number of variables does not exceed the allowed variable capacity. Next, we provide some information on this technology.

4.2. The Digital Annealer

The Digital Annealer (DA) is a quantum-inspired computer architecture designed to derive solutions for combinatorial optimization problems formulated as a QUBO model. It consolidates the merits of both quantum and general-purpose computers, and takes advantage of the massive parallelization that its hardware allows (Aramon et al., 2019; Sao et al., 2019). The first generation of DA is capable of solving problems with up to 1024 variables, while this number has increased to 8192 in the second generation (Fujitsu Limited, 2020b).

The algorithm of DA is based on simulated annealing. Simulated annealing (SA) is a probabilistic method for finding solutions to combinatorial optimization problems that aim to minimize some cost function, by making an analogy to the physical process of *annealing* whereby a heated material is slowly cooled until it reaches a state of minimum energy (Kirkpatrick et al., 1983; Bertsimas et al., 1993). The idea in simulated annealing is to propose a random perturbation to the current solution at each iteration, evaluate the consequent change in the objective function, and decide whether or not to move to the proposed solution. If the proposed solution results in a lower objective value, it is always accepted; otherwise, i.e., if it is a "uphill" move, it is accepted with a probability that is a function of the change in the objective value and the current temperature. While higher temperatures more likely permit uphill moves to let the algorithm explore a larger region of the objective function and to help escape from local optima, the search intensifies around a narrower area with lower temperatures. Under certain conditions, simulated annealing asymptotically converges to a global optimum, yet, it may necessitate infinitely many iterations. So, in practice, it is very well possible to converge to a local optimum in simulated annealing (Kirkpatrick et al., 1983; Rutenbar, 1989; Glover and Kochenberger, 2006; Gendreau et al., 2010).

To apply simulated annealing based algorithms, one needs to define a *solution representation* as well as a *move* operation to propose a new candidate solution at each iteration (Kirkpatrick et al., 1983). In DA, a solution (to a QUBO problem) is represented with a vector of binary variable values, and the move operation is defined as the *flip* of a variable value, i.e., changing the value of a variable from one to zero or vice versa.

While being grounded in simulated annealing, DA's algorithm differs from it in some key aspects. First, it uses a *parallel trial* scheme, where it evaluates all possible moves in parallel at each iteration, as opposed to the classical way of considering one random move only. When more than one flip is eligible for acceptance, one of them is chosen uniformly at random. Second, it utilizes a *dynamic offset* mechanism to escape from local optima, such that if no flip is accepted in the current iteration, the acceptance probabilities in the subsequent iteration are artificially increased. Specifically, when no candidate variable to flip can be found, a positive offset value is added to the objective function, equivalent to multiplying the acceptance probabilities with a coefficient that is a function of the current temperature and the magnitude of the offset. Otherwise, the offset value is set to zero. Third, DA has the *parallel tempering* option, also referred to as the *replica exchange* method, where multiple independent search processes (replicas) are initiated in parallel with a different temperature each, and states (solutions) are probabilistically exchanged between them. This way, each replica performs a random walk in the temperature space, helping to avoid being stuck at a local minimum (Aramon et al., 2019; Hukushima and Nemoto, 1996; Matsubara et al., 2020). In our computational experiments, we utilize DA in parallel tempering mode.

5. Computational Study

In this section, we present the results of our computational study. We generated a large suite of RWA-P instances using known networks from the literature and conducted detailed analysis. Our experimental setting can be summarized as follows:

- **Solvers.** We used the second generation of DA[1] (Matsubara et al., 2020) and GUROBI 9.0[2] (Gurobi Optimization LLC, 2020).

- **Methods.** While we (1) provided the QUBO formulation to DA, we (2) employed GUROBI in three different ways; (i) to directly solve the IP formulations, both IP$^{\text{Base}}$ and IP$^{\text{Strong}}$, (ii) to solve IP$^{\text{Strong}}$ via branch-and-cut (B&C) using the lazy callback feature, and (iii) to directly solve the QUBO formulation. We note that we sometimes refer to (i) as "GUROBI as IP solver", and to (iii) as "GUROBI as QUBO solver". In addition, we solve RWA-P via the random-search-based heuristic from (Ezzahdi et al., 2006), which was also utilized in (Azodolmolky et al., 2010). We refer to this heuristic as RS-Heur.

- **Time limit.** We used three different time limits; 120 seconds, which is approximately the highest run time DA takes for our particular problem, 600 seconds to compare the longer run-time performance of GUROBI to that of DA, and 7200 seconds (two hours) to see how well GUROBI can achieve in cases where hours of run times are tolerable.

- **Experiments.** We carried out three main groups of analyses; (1) performance comparison of the five alternative methods, (2) effect of using solutions from DA as an initial solution for

[1] For DA experiments, we used the Digital Annealer environment prepared exclusively for research at the University of Toronto.

[2] For GUROBI experiments, we used a MacOS computer with 3 GHz Intel Core i5 CPU and 16 GB memory.

GUROBI, as well as a run time analysis for GUROBI to reach DA's performance level, and (3) sensitivity analysis of DA's performance to values of the penalty coefficient ρ.

- **Implementation details.** First, we implemented a B&C algorithm using the callback features of GUROBI to pass the conflict constraints as needed. We tried various cut selection and management strategies for user and lazy callbacks, the best of which yielded no better performance than using the lazy callbacks, i.e. by adding a conflict constraint each time a feasible solution violating it is encountered, which, therefore, is the one we utilize in our experiments. Second, although it is not possible to explicitly impose a time limit for DA, after some preliminary testing, we set the number of iterations to an appropriate value yielding the desired execution time of 120 seconds, which indeed was the highest time we observed for our particular problem, RWA-P. Once the number of iterations are fixed, DA's execution times show almost no variability across different instances. Third, we failed to utilize the penalty coefficient values suggested in Proposition 4 because the values were outside the acceptable range for DA. Thus, we used smaller values for the penalty coefficient that do not necessarily guarantee the exactness of the model, but always yielded feasible solutions in practice.

The remainder of this section is organized as follows. In Section 5.1, we provide some details about the networks used and the way we generated our problem instances. In Section 5.2, we present our main set of experimental results, follow it by an initial solution and run time analysis in Section 5.3, and then provide the results of our penalty coefficient analysis in Section 5.4. Finally in Section 5.5, we summarize our key observations as a result of our computational study.

5.1. Problem Instances

In order to generate our test instances, we made use of four different network topologies of varying sizes and densities from the literature, namely EON (Tornatore et al., 2007), Brazil (Jaumard et al., 2006), USA and China (Hwang et al., 2009). In Table 1, we provide descriptive information about the networks and the test instances we generated from them.

For all the networks, we assume that each edge $\{u, v\}$ is represented with a pair of links (u, v) and (v, u). We use three different wavelength capacities on links, $|\Lambda| \in \{5, 10, 15\}$, and three different numbers for requests, $|R| \in \{60, 80, 100\}$. We generate five random instances for each parameter combination, which makes a total of 180 test instances. For each instance, we selected a distinct source and destination pair for every request among all possible ordered node pairs in the network. For each request, we randomly selected four working and four protection paths between the source

Table 1: Instance information.

	Network features			Instance parameters		Instance sizes		
Network	# nodes	# links	Avg	# wavelengths	# requests	# vars	# cons / # vars	# cons / # vars
	$(\lvert V\rvert)$	$(\lvert E\rvert)$	in/out-degree	$(\lvert\Lambda\rvert)$	$(\lvert R\rvert)$		in IP$^{\text{Strong}}$	in IP$^{\text{Base}}$
EON	19	78	2.05	{5, 10, 15}	{60, 80, 100}	[2400, 8000]	0.67	234.0
USA	24	86	1.79	{5, 10, 15}	{60, 80, 100}	[2400, 8000]	0.69	212.0
Brazil	27	140	2.67	{5, 10, 15}	{60, 80, 100}	[2400, 8000]	0.77	131.3
China	39	144	1.85	{5, 10, 15}	{60, 80, 100}	[2400, 8000]	0.79	174.5

and destination nodes, except when $|\Lambda| = 15$ and $|R| \in \{80, 100\}$ where we decreased the number of working/protection paths to three and two, so that the sizes of those instances become eligible for DA, which can handle at most 8192 variables. We formed the working and protection lightpaths of every request by combining each generated path with every one of the available wavelengths.

As indicators of instance sizes, we provide the number of variables ("# vars") as well as the ratio of the number of constraints to the number of variables ("# cons / # vars"). both for IP$^{\text{Strong}}$ and IP$^{\text{Base}}$ formulations. It is noteworthy that the strengthened set of our conflict constraints leads to a remarkable decrease in the total number of constraints; the ratio of the number of constraints to the number of variables is 170 to 350 times higher in IP$^{\text{Base}}$ than in IP$^{\text{Strong}}$.

5.2. Performance Comparison of the Methods

In this section, we report the results of our main set of experiments and compare the performances of all the methods under consideration, including RS-Heur. For our particular problem RWA-P, the steps of RS-Heur are as follows: At each step, we generate a random permutation of the request set. Considering the requests in the order they appear in the permutation, we first find a link-disjoint pair of working and protection paths, starting the search with the shortest paths in the precomputed set of paths for the associated request. If such a pair exists, we search for the first available wavelength whose assignment does not lead to a conflict with the set of lightpaths assigned to the current set of granted requests, if any. If no wavelength is available, we select the next pair of working and protection paths from the precomputed set (following an increasing order in path lengths), and seek the first available wavelength as before. This procedure continues until we find a feasible pair of lightpaths or no alternative is left to consider. In our implementation, we continue considering different permutations of the set of requests until the imposed time limit is reached, and a 120-second time limit yielded an average of 7243 permutations per instance in our test set. At the end, we select a solution with maximum number of granted requests over the set of all considered permutations.

Table 2 reports the number of granted requests for all methods averaged over all wavelength and request numbers, and the average number of links used per lightpath of a granted request in parentheses in a second row for each network, under two groups of columns corresponding to the experiments with 120- and 600-second time limits. For direct solving of IP$^{\text{Strong}}$ with GUROBI, we present additional sets of results obtained by setting the solver's emphasis on search for feasible solutions ("IP$^{\text{Strong}}$ (f)") for both time limits, while all other GUROBI-related results are obtained under default settings.

Table 2: Summary of overall results in terms of the number of granted requests (average number of links per granted request shown in parentheses).

	120 sec							600 sec	
			GUROBI					GUROBI	
Network	DA	RS-Heur	IP$^{\text{Strong}}$ (f)	IP$^{\text{Strong}}$	IP$^{\text{Base}}$	B&C	QUBO	IP$^{\text{Strong}}$ (f)	IP$^{\text{Strong}}$
EON	23.3	13.7	21.0	20.6	16.7	2.2	6.1	21.6	21.0
	(9.1)	(9.4)	(9.0)	(9.2)	(9.3)	(9.0)	(8.8)	(9.0)	(9.0)
USA	24.7	15.3	22.6	22.2	19.9	2.3	5.0	23.2	22.7
	(8.9)	(9.0)	(8.9)	(9.1)	(8.9)	(8.5)	(8.9)	(8.9)	(8.9)
Brazil	43.7	28.1	37.7	38.1	30.8	7.7	7.6	41.0	40.0
	(8.1)	(8.0)	(8.1)	(8.2)	(8.1)	(7.8)	(7.5)	(8.1)	(8.1)
China	31.6	19.4	28.4	28.2	22.9	2.1	5.1	29.2	28.7
	(10.4)	(10.2)	(10.3)	(10.5)	(10.3)	(10.1)	(9.7)	(10.3)	(10.4)

We observe from Table 2 that the number of granted requests obtained from DA in 120 seconds outperform those obtained from 120- and even 600-second experiments with GUROBI for all networks. Among the GUROBI-based alternatives, on the other hand, solving of IP$^{\text{Strong}}$ directly with GUROBI yields the best results, most of the time slightly better when the emphasis is on finding feasible solutions. Even though the results from RS-Heur are better than those obtained from solving IP$^{\text{Strong}}$ through B&C (with callback) and the QUBO formulation directly with GUROBI, they are all still far from being comparable to the other alternatives. Also, IP$^{\text{Strong}}$ yields considerably better results than IP$^{\text{Base}}$, as expected, and increasing the time limit from 120 to 600 seconds leads to a relative improvement of only 0.4 to 1.9 in the average number of granted requests for IP$^{\text{Strong}}$. As for the link usage, since it is our secondary objective, two solutions with different numbers of granted requests cannot be compared on the basis of the total number of links used. Nevertheless, in terms of the average number of links used per lightpath of a granted request, DA in 120 seconds and GUROBI in 600 seconds perform almost always the same, even though DA achieves a higher number of granted requests.

One might expect the B&C method to compete with feeding the IP formulation to the solver as a whole; however, B&C performs even worse than directly solving the QUBO formulation with GUROBI mostly. In order to see whether it could provide good-quality solutions when longer run times are allowed, we also tested it with a 600-second time limit on a selected set of 30 instances based on USA and Brazil networks, with $(|\Lambda|, |R|) \in \{(5, 60), (10, 80), (15, 100)\}$, where $|\Lambda|$ denotes the number of wavelengths and $|R|$ the number of requests. For each pair of parameter values, we have ten instances in this selected set, five per network. For half of these instances based on USA network, the average number of granted requests rises from 3.3 to 7.5, and for the remaining ones based on Brazil network it rises from 4.8 to 10.2. Consequently, we conclude that the B&C method fails to be a favorable method for further consideration, as do solving IP^{Base} and QUBO formulations directly with GUROBI. We also tested the heuristic RS-Heur under a 600-second time limit, and observed that the increase in the average number of granted requests is less than one, confirming that this method too is not a favorable option for further consideration.

We confine the rest of our analysis to the most promising two alternatives; namely, solving $\text{IP}^{\text{Strong}}$ using GUROBI with an emphasis on finding feasible solutions, and solving the QUBO formulation with DA. Figure 2 compares the average number of granted requests for every $(|\Lambda|, |R|)$ pair, using 120- and 600-second time limits for DA and GUROBI, respectively. The four plots demonstrate that DA outperforms GUROBI for each parameter combination and every one of the four networks, and the difference is particularly evident when the number of wavelengths is ten. In fact, DA not only outperforms in terms of the average values but also provides predominantly superior or otherwise as good results in almost all individual instances. These results show that for our test set comprising instances that are hard to optimally solve, DA yields solutions within only two minutes which are better than or as good as the ones GUROBI attains in ten minutes.

5.3. Initial Solution and Run Time Analysis

Having observed that GUROBI cannot much improve the solution quality despite the significant increase in the time limit, we now investigate whether feeding a DA solution to GUROBI as an initial (warm start) solution would improve its performance, and also how much time it would take for GUROBI to reach DA's performance level. To this end, we use the set of 30 representative instances mentioned above, which covers both the extremes and the intermediate portion of the

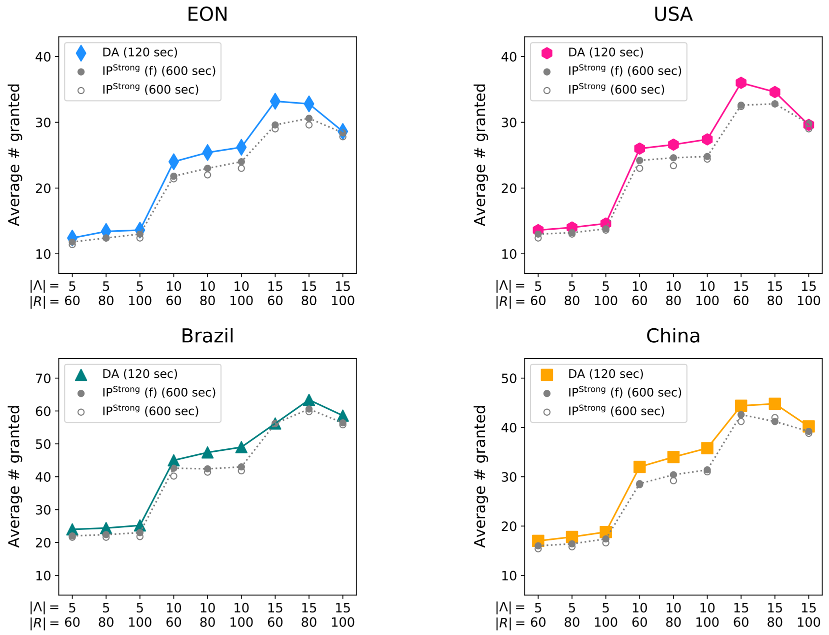

Figure 2: Comparison of 120-second DA and 600-second IP$^{\text{Strong}}$ results (with and without the solver's emphasis on finding feasible solution for the latter) for every combination of number of wavelengths and requests.

parameter space. For our initial solution analysis, we feed the DA solutions to GUROBI as a starting point in solving IP$^{\text{Strong}}$ with a time limit of 600 seconds and emphasis on finding feasible solutions. In order to look into the possible effect of incorporating link usage in the objective, we additionally carry out the same set of experiments by maximizing the number of granted requests only, i.e., by setting $\alpha = 0$ and $\beta = 1$. For the run time analysis, we set the objective values of solutions from 120-second DA experiments as a stopping condition for GUROBI, along with a time limit of 7200 seconds (two hours). Note that the objective value of a solution with a certain number of granted requests cannot be attained or surpassed without granting at least that many requests due to the way we set the value of the β coefficient for prioritizing request granting over link usage.

Table 3 contains the results of the above-mentioned experiments. The results of the initial solution experiments are provided under a five-column block ("IP$^{\text{Strong}}$ (f) w/ initial sol (600 sec)"), and is divided into two groups; one for the results of the experiments with the original objective function that incorporates both the request granting and link usage components ("Weighted obj"), and the other for the case when only the request granting component of the objective is considered ("Request granting obj"). Each group contains the number of granted requests and used links averaged over the five instances associated with the parameter pair $(|\Lambda|, |R|)$ and network in each row. For the experiments with the original weighted objective, average optimality gap is also reported, defined as % gap = $\frac{\text{UB}-\text{LB}}{|\text{UB}|} \cdot 100$ where UB and LB respectively denote the upper and lower bounds for the objective value. The last block of columns contains the results of the run time analysis experiments ("$\text{IP}^{\text{Strong}}$ (f) until DA obj val (7200 sec)"). This block additionally includes the average run times in the last column ("Time"), as well as the number of granted requests GUROBI attains when the experiments are allowed to run for a full two hours, i.e., when the objective level of DA is not imposed as a stopping condition, provided in parentheses.

Table 3: Initial solution and run time analysis results for the selected set of representative instances.

				$\text{IP}^{\text{Strong}}$ (f) w/ initial sol (600 sec)					$\text{IP}^{\text{Strong}}$ (f) until DA obj val (7200 sec)			
		DA (120 sec)		Weighted obj			Request granting obj		Weighted obj			
$(\lvert\Lambda\rvert, \lvert R\rvert)$	Network	# granted	# links	# granted	# links	% gap	# granted	# links	# granted	# links	% gap	Time
$(5, 60)$	USA	13.6	227.8	13.6	227.6	30.6	13.6	227.8	13.4 (13.4)	226.6	31.1	4343.8
	Brazil	24.0	379.0	24.0	379.0	50.4	24.0	379.0	22.8 (22.8)	357.0	57.6	6742.7
$(10, 80)$	USA	26.6	485.0	26.6	485.0	35.1	26.6	485.0	25.2	443.2	42.1	7200.0
	Brazil	47.4	777.6	47.4	777.0	47.8	47.6	781.6	42.8	706.2	63.7	7200.0
$(15, 100)$	USA	29.6	589.8	30.2	600.8	12.0	30.2	606.2	29.6 (30.2)	587.4	15.8	222.8
	Brazil	58.6	985.8	58.6	983.4	17.6	58.6	985.8	57.2 (57.8)	950.6	20.6	5556.8

We see from Table 3 that GUROBI is mostly unable to improve the initial solution from DA within the 600-second time limit, both in terms of granted requests and link usage. The situation remains the same when the objective is reduced into the maximization of request granting, indicating that the link usage component in the objective does not really affect the performance, and thus a hierarchical solution approach for RWA-P to handle the two separate objectives would not perform any better than solving the problem with a weighted objective. As for the run time analysis, in 66.7% of the selected instances (20 out of 30), GUROBI cannot reach the 120-second objective value of DA after two hours. The time for the remaining 10 instances to reach DA's objective value is 1233 seconds on the average, an order of magnitude higher than DA's run time approximately. Also, as a result of the full two-hour experiments, the average number of granted requests (shown in parentheses) either does not change or increases by 0.6 only. In terms of average optimality gaps, the values from the initial solution experiments are 6.2% better than those of the run time experiments. This is mainly due to the solutions from DA, which yield 4.3% better upper bounds than the ones from the run time experiments, while lower bounds are only 0.1% worse. These results indicate that DA is capable of delivering good-quality solutions in only two minutes, which typically cannot be attained by a state-of-the-art solver after two hours of run time.

5.4. *Penalty Coefficient Analysis*

As mentioned before, penalty coefficient values rendering our QUBO formulations exact go beyond the acceptable ranges for DA. Nevertheless, we observed in our preliminary experiments that by setting the penalty coefficient values with respect to the β values (which we compute using Proposition 2), DA can always deliver solutions that are feasible for the IP models. Specifically, we set $\rho = \beta + 100$, yielding IP feasible solutions throughout all of our experiments. While testing different values for the penalty coefficient, however, we noticed that it can have a considerable impact on DA's performance, which we investigate in the sequel.

Figure 3 contains the results from 120-second DA experiments with different penalty coefficient values, using the same set of 30 representative instances mentioned above. While x-axes of the two plots show the penalty coefficient values as a function of the β parameter, y-axes display the average number of granted requests. The lower ends of the penalty coefficient values are those used in our main set of experiments, whereas the higher end is set close to the values before passing beyond the acceptable ranges for DA.

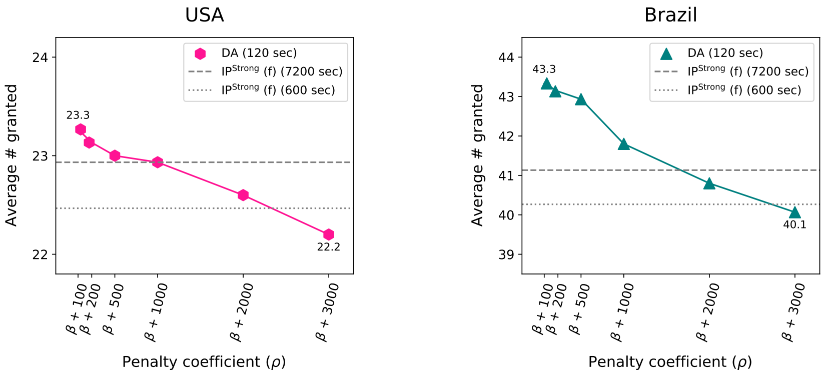

Figure 3: Results of the penalty coefficient trials with DA using the selected set of representative instances.

We observe that decreasing the penalty coefficient values improve DA's performance, and the difference is more marked for the instances based on Brazil network. In comparison to the two-hour performance of GUROBI, not only does DA outperform with the best possible penalty coefficient values for it (i.e., when $\rho = \beta + 100$), it yields better or at least as good results when $\beta + 100 \leq \rho \leq \beta + 1000$. Thus, even without much tuning of the penalty coefficient parameter, DA in two minutes delivers better results than a state-of-the-art solver does after hours of run times.

5.5. *Key Observations*

We lastly summarize some main findings arising from our computational study.

- DA is the best-performing option for RWA-P to obtain high-quality solutions in a short amount of time for our nontrivial set of test instances, and established methods coupled with state-of-the-art solvers yield mostly inferior results even after two hours of run time.

- DA's performance is robust across different instances and networks.

- Feeding solutions obtained from DA as initial solutions to GUROBI does not improve its performance; it cannot find any better solution even after a considerable amount of run time.

- Solving RWA-P by only considering the prioritized objective of request granting and ignoring the secondary one (link usage) does not perform any better than solving it with a weighted objective.

- Penalty coefficient values affect DA's solution quality, but even without particularly tuned values DA can outperform the two-hour results from the state of the art.

6. Conclusion

In this study, we consider the routing and wavelength assignment problem with protection, RWA-P. Through complexity analysis and computational experiments, we show that this problem is difficult to solve both in theory and practice. We propose a viable approach, formulating RWA-P as a QUBO model and employing the Digital Annealer, DA, as a new promising solution technology. We find that this approach outperforms established methods in handling our nontrivial set of test insances by delivering solutions within two minutes that are superior to or as good as the ones other methods yield in two hours, as well as a heuristic method from the literature. Also considering that future generations of DA are planned to achieve megabit-class performance (Fujitsu Limited, 2020a), we believe that the proposed approach has significant potential to be utilized widely in practice. As such, future research directions involve considering large-scale cases of RWA-P, adaptation and application of this emerging approach to other RWA problems.

Acknowledgement

The authors would like to thank Fujitsu Laboratories Ltd. and Fujitsu Consulting (Canada) Inc. for providing financial support and access to Digital Annealer at the University of Toronto.

References

Aramon, M., Rosenberg, G., Valiante, E., Miyazawa, T., Tamura, H., Katzgraber, H., 2019. Physics-inspired optimization for quadratic unconstrained problems using a digital annealer. Frontiers in Physics 7, 48.

Azodolmolky, S., Klinkowski, M., Pointurier, Y., Angelou, M., Careglio, D., Solé-Pareta, J., Tomkos, I., 2010. A novel offline physical layer impairments aware rwa algorithm with dedicated path protection consideration. Journal of Lightwave Technology 28 (20), 3029–3040.

Bandyopadhyay, S., 2007. Dissemination of Information in Optical Networks: From Technology to Algorithms. Springer Science & Business Media.

Bertsimas, D., Tsitsiklis, J., et al., 1993. Simulated annealing. Statistical science 8 (1), 10–15.

Boyd, J., 2018. Silicon chip delivers quantum speeds [news]. IEEE Spectrum 55 (7), 10–11.

Chadha, D., 2019. Optical WDM Networks: From Static to Elastic Networks. Wiley Online Library.

Chatterjee, B. C., Sarma, N., Sahu, P. P., Oki, E., 2016. Routing and Wavelength Assignment for WDM-based Optical Networks: Quality-of-Service and Fault Resilience. Vol. 410. Springer.

Chiu, A. L., Modiano, E. H., 2000. Traffic grooming algorithms for reducing electronic multiplexing costs in WDM ring networks. Journal of Lightwave Technology 18 (1), 2–12.

Chlamtac, I., Ganz, A., Karmi, G., 1992. Lightpath communications: An approach to high bandwidth optical WAN's. IEEE Transactions on Communications 40 (7), 1171–1182.

Cohen, E., Mandal, A., Ushijima-Mwesigwa, H., Roy, A., 2020a. Ising-based consensus clustering on specialized hardware. In: International Symposium on Intelligent Data Analysis. Springer, pp. 106–118.

Cohen, E., Senderovich, A., Beck, J. C., 2020b. An Ising framework for constrained clustering on special purpose hardware. In: International Conference on the Integration of Constraint Programming, Artificial Intelligence, and Operations Research. To appear.

Corder, K., Monaco, J. V., Vindiola, M. M., 2018. Solving vertex cover via Ising model on a neuromorphic processor. In: 2018 IEEE International Symposium on Circuits and Systems (ISCAS). IEEE, pp. 1–5.

Ebrahimzadeh, A., Rahbar, A. G., Alizadeh, B., 2013. Binary quadratic programming formulation for routing and wavelength assignment problem in all-optical WDM networks. Optical Switching and Networking 10 (4), 354–365.

Erlebach, T., Jansen, K., 2001. The complexity of path coloring and call scheduling. Theoretical Computer Science 255 (1-2), 33–50.

Ezzahdi, M. A., Al Zahr, S., Koubàa, M., Puech, N., Gagnaire, M., 2006. Lerp: a quality of transmission dependent heuristic for routing and wavelength assignment in hybrid wdm networks. In: Proceedings of 15th International Conference on Computer Communications and Networks. IEEE, pp. 125–136.

Fawaz, W., Daheb, B., Audouin, O., Du-Pond, M., Pujolle, G., 2004. Service level agreement and provisioning in optical networks. IEEE Communications Magazine 42 (1), 36–43.

Fujitsu Limited, 2020a. Fujitsu develops new tech for quantum-inspired "Digital Annealer", achieving megabit-class performance for large-scale combinatorial optimization problems. https://www.fujitsu.com/global/about/resources/news/press-releases/2020/1109-01.html.

Fujitsu Limited, 2020b. Fujitsu launches next generation quantum-inspired Digital Annealer service. https://www.fujitsu.com/global/about/resources/news/press-releases/2018/1221-01.html.

Garey, M. R., Johnson, D. S., 1979. Computers and Intractability: A Guide to the Theory of NP-Completeness. Freeman, San Francisco.

Gendreau, M., Potvin, J.-Y., et al., 2010. Handbook of metaheuristics. Vol. 2. Springer.

Giovanni, L. D., 2017. Heuristics for combinatorial optimization. Course pack for METMODOC: Methods and Models for Combinatorial Optimization, University of Padua, https://www.math.unipd.it/~luigi/courses/metmodoc1819/m02.meta.en.partial01.pdf Accessed 4 August 2020.

Glover, F. W., Kochenberger, G. A. (Eds.), 2006. Handbook of metaheuristics. Vol. 57. Kluwer.

Grover, W. D., May 7 2013. Distributed synchronous batch reconfiguration of a network. US Patent 8,437,280.

Gurobi Optimization LLC, 2020. Gurobi Optimizer reference manual. https://www.gurobi.com/wp-content/plugins/hd_documentations/documentation/9.0/refman.pdf.

Hukushima, K., Nemoto, K., 1996. Exchange Monte Carlo method and application to spin glass simulations. Journal of the Physical Society of Japan 65 (6), 1604–1608.

Hwang, I.-S., Lee, S.-N., Shyu, Z.-D., Chen, K.-P., 2009. One-to-many multicast restoration based on dynamic core-based selection algorithm in wdm mesh networks. Photonic Network Communications 18 (3), 275–286.

IBM, 2019. ILOG CPLEX Studio 12.9 manual.

Inagaki, T., Haribara, Y., Igarashi, K., Sonobe, T., Tamate, S., Honjo, T., Marandi, A., McMahon, P. L., Umeki, T., Enbutsu, K., et al., 2016. A coherent Ising machine for 2000-node optimization problems. Science 354 (6312), 603–606.

Jaumard, B., Meyer, C., Yu, X., 2006. How much wavelength conversion allows a reduction in the blocking rate? Journal of Optical Networking 5 (12), 881–900.

Javad-Kalbasi, M., Dabiri, K., Valaee, S., Sheikholeslami, A., 2019. Digitally annealed solution for the vertex cover problem with application in cyber security. In: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, pp. 2642–2646.

Kirkpatrick, S., Gelatt, C. D., Vecchi, M. P., 1983. Optimization by simulated annealing. Science 220 (4598), 671–680.

Kokkinos, P., Manousakis, K., Varvarigos, E., 2010. Path protection in WDM networks with quality of transmission limitations. In: 2010 IEEE International Conference on Communications. pp. 1–5.

Lee, T., Park, S., 2006. Comparison of wavelength requirements between two wavelength assignment methods in survivable WDM networks. Annals of Operations Research 146 (1), 75–89.

Li, G., Simha, R., 2000. The partition coloring problem and its application to wavelength routing and assignment. In: Proceedings of the First Workshop on Optical Networks. Citeseer, p. 1.

Li Shifeng, Tao Jun, Gu Guanqun, 2002. Routing and wavelength assignment in all optical networks to establish survivable lightpaths. In: 2002 IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering. TENCOM '02. Proceedings. Vol. 2. pp. 1193–1196 vol.2.

Losego, F., Tornatore, M., Maier, G., Pattavina, A., 2005. Time constraints in an OTN semi-automatic control system. In: Proceedings of 5th International Workshop on Design of Reliable Communication Networks. IEEE, pp. 7–14.

Majumdar, A. K., 2018. Optical wireless communications for broadband global internet connectivity: Fundamentals and potential applications. Elsevier.

Matsubara, S., Takatsu, M., Miyazawa, T., Shibasaki, T., Watanabe, Y., Takemoto, K., Tamura, H., 2020. Digital Annealer for high-speed solving of combinatorial optimization problems and its applications. In: 2020 25th Asia and South Pacific Design Automation Conference (ASP-DAC). IEEE, pp. 667–672.

Mukherjee, B., 2006. Optical WDM networks. Springer Science & Business Media.

Naghsh, Z., Javad-Kalbasi, M., Valaee, S., 2019. Digitally annealed solution for the maximum clique problem with critical application in cellular v2x. In: ICC 2019-2019 IEEE International Conference on Communications (ICC). IEEE, pp. 1–7.

Noronha, T. F., Ribeiro, C. C., 2006. Routing and wavelength assignment by partition colouring.

European Journal of Operational Research 171 (3), 797–810.

Ohzeki, M., Miki, A., Miyama, M. J., Terabe, M., 2019. Control of automated guided vehicles without collision by quantum annealer. Frontiers in Computer Science 1, 9.

Papalitsas, C., Andronikos, T., Giannakis, K., Theocharopoulou, G., Fanarioti, S., 2019. A QUBO model for the traveling salesman problem with time windows. Algorithms 12 (11), 224.

Rahman, M. T., Han, S., Tadayon, N., Valaee, S., 2019. Ising model formulation of outlier rejection, with application in wifi based positioning. In: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, pp. 4405–4409.

Ramamurthy, S., Sahasrabuddhe, L., Mukherjee, B., 2003. Survivable WDM mesh networks. Journal of Lightwave Technology 21 (4), 870.

Rutenbar, R. A., 1989. Simulated annealing algorithms: An overview. IEEE Circuits and Devices Magazine 5 (1), 19–26.

Sao, M., Watanabe, H., Musha, Y., Utsunomiya, A., 2019. Application of Digital Annealer for faster combinatorial optimization. Fujitsu Scientific & Technical Journal 55 (2), 45–51.

Şeker, O., Tanoumand, N., Bodur, M., 2020. Digital annealer for quadratic unconstrained binary optimization: a comparative performance analysis. arXiv preprint arXiv:2012.12264.

Shen, L., Yang, X., Ramamurthy, B., 2005. Shared risk link group (SRLG)-diverse path provisioning under hybrid service level agreements in wavelength-routed optical mesh networks. IEEE/ACM Transactions on Networking 13 (4), 918–931.

Tornatore, M., Maier, G., Pattavina, A., 2007. Wdm network design by ilp models based on flow aggregation. IEEE/ACM Transactions On Networking 15 (3), 709–720.

Wang, Y., Li, L., Wang, S., 2001. A new algorithm of design protection for wavelength-routed networks and efficient wavelength converter placement. In: IEEE International Conference on Communications. Vol. 6. pp. 1807–1811.

Wu, J., Zhang, J., von Bochmann, G., Savoie, M., 2012. Forward-looking WDM network reconfiguration with per-link congestion control. Journal of Network and Systems Management 20 (1), 6–33.

Zhang, J., Mukheriee, B., 2004. A review of fault management in WDM mesh networks: Basic concepts and research challenges. IEEE Network 18 (2), 41–48.

Zhang, J. Y., Yang, O. W., Wu, J., Savoie, M., 2007. Optimization of semi-dynamic lightpath rearrangements in a WDM network. IEEE Journal on Selected Areas in Communications 25 (9), 3–17.

Appendix A. Proofs

In this section, we provide the proofs of all the propositions, lemmas and theorems we present in the paper, as well as their statements for the sake of completeness. We number the equations in this section with a prefix of "A.".

Appendix A.1. Proof of Proposition 1

The optimization models in (3) *and* (4) *yield the same optimal values; that is,* $\Omega^{>} = \Omega^{=}$*. This implies that the prioritization condition given in* (3) *can also be achieved by setting* $\alpha, \beta > 0$ *such that* $\frac{\beta}{\alpha} > \Omega^{=}$*.*

Proof. We first show that $\Omega^{=} \geq \Omega^{>}$, indicating that the sufficiency of the lower bound from model (3) is implied by that of (4), which can be expressed as the following logical relationship:

$$\frac{\beta}{\alpha} \; > \; \Omega^{=} \quad \Longrightarrow \quad \frac{\beta}{\alpha} \; > \; \Omega^{>}. \tag{A.1}$$

Suppose that the left-hand side of (A.1) holds, which means that for any pair of feasible solutions $(\hat{x}, \hat{y})$ and $(\tilde{x}, \tilde{y})$ with $\hat{f}_\beta = \tilde{f}_\beta + 1$, the prioritization condition given in (2) holds, yielding

$$\alpha \hat{f}_\alpha^{\max} - \beta \hat{f}_\beta \; < \; \alpha \tilde{f}_\alpha^{\min} - \beta \tilde{f}_\beta. \tag{A.2}$$

Let $(\bar{x}, \bar{y})$ be a feasible solution with $\bar{f}_\beta = \hat{f}_\beta + 1$. Then, the following holds by assumption:

$$\alpha \bar{f}_\alpha^{\max} - \beta \bar{f}_\beta \; < \; \alpha \hat{f}_\alpha^{\min} - \beta \hat{f}_\beta. \tag{A.3}$$

Combining (A.2) and (A.3), we get

$$\alpha \bar{f}_\alpha^{\min} - \beta \bar{f}_\beta \; < \; \alpha \bar{f}_\alpha^{\max} - \beta \bar{f}_\beta \; < \; \alpha \hat{f}_\alpha^{\max} - \beta \hat{f}_\beta \; < \; \alpha \tilde{f}_\alpha^{\min} - \beta \tilde{f}_\beta. \tag{A.4}$$

This shows that $\bar{f}_\beta = \tilde{f}_\beta + 2 \implies \bar{f} < \tilde{f}$. Since the chain of inequalities in (A.4) can be extended to any feasible solution $(\bar{x}, \bar{y})$ with $\bar{f}_\beta = \tilde{f}_\beta + k$ for $k \geq 2$, we conclude that the relationship in (A.1) holds and hence $\Omega^{=} \geq \Omega^{>}$. Moreover, the feasible region of the model in (3) contains that of (4), thus $\Omega^{=} \leq \Omega^{>}$, which establishes $\Omega^{>} = \Omega^{=}$. □

Appendix A.2. Proof of Proposition 2

Selecting $\alpha, \beta > 0$ *such that*

$$\frac{\beta}{\alpha} \; > \; |R| \; (M - 2) \; + \; 2 \tag{5}$$

prioritizes request granting over link usage in (1a) *for any feasible solution to the IP, i.e., solutions accepting more requests yield lower objective values, where* $M = \max_{r \in R} \left\{ \max_{w \in W^r} \left\{ B_w^r \right\} + \max_{p \in P^r} \left\{ B_p^r \right\} \right\}$*.*

Proof. Suppose that the parameters $\alpha, \beta > 0$ satisfy condition (5). Let $(\hat{x}, \hat{y})$ and $(\tilde{x}, \tilde{y})$ be two feasible solutions that satisfy $\hat{f}_\beta > \tilde{f}_\beta$. By Proposition 1, it suffices to show that the prioritization condition in (2) holds when $\hat{f}_\beta = \tilde{f}_\beta + 1$, i.e., we want to show the following logical relationship:

$$\hat{f}_\beta \ = \ \tilde{f}_\beta + 1 \implies \alpha \hat{f}_\alpha^{\max} - \beta \hat{f}_\beta \ < \ \alpha \tilde{f}_\alpha^{\min} - \beta \tilde{f}_\beta \tag{A.5}$$

Assume for a contradiction that we have $\hat{f}_\beta = \tilde{f}_\beta + 1$ and $\alpha \hat{f}_\alpha^{\max} - \beta \hat{f}_\beta \ \geq \ \alpha \tilde{f}_\alpha^{\min} - \beta \tilde{f}_\beta$, which can be equivalently written as

$$\beta \hat{f}_\beta \ - \ \beta \tilde{f}_\beta \ \leq \ \alpha \hat{f}_\alpha^{\max} \ - \ \alpha \tilde{f}_\alpha^{\min}.$$

Rearranging the terms, we obtain

$$\frac{\beta}{\alpha} \ \leq \ \frac{\hat{f}_\alpha^{\max} \ - \ \tilde{f}_\alpha^{\min}}{\hat{f}_\beta \ - \ \tilde{f}_\beta} \ = \ \hat{f}_\alpha^{\max} \ - \ \tilde{f}_\alpha^{\min}, \tag{A.6}$$

because $\alpha > 0$ and $\hat{f}_\beta - \tilde{f}_\beta = 1$. Since each lightpath is comprised of at least one link, we have

$$\tilde{f}_\alpha^{\min} \ \geq \ 2\tilde{f}_\beta. \tag{A.7}$$

Moreover, in the worst case, the longest working and protection lightpaths will be used for each granted request, yielding

$$\hat{f}_\alpha^{\max} \ \leq \ \hat{f}_\beta \cdot \max_{r \in R} \left\{ \max_{w \in W^r} \left\{ B_w^r \right\} \ + \ \max_{p \in P^r} \left\{ B_p^r \right\} \right\} \ =: \ \tilde{f}_\beta \ M. \tag{A.8}$$

Combining (A.6), (A.7) and (A.8), we obtain

$$\frac{\beta}{\alpha} \ \leq \ \hat{f}_\alpha^{\max} \ - \ \tilde{f}_\alpha^{\min} \ \leq \ \hat{f}_\beta \ M \ - \ 2\tilde{f}_\beta.$$

Plugging in $\hat{f}_\beta - 1 = \tilde{f}_\beta$, we get

$$\frac{\beta}{\alpha} \ \leq \ \hat{f}_\beta \ M \ - \ 2(\hat{f}_\beta - 1) \ = \ \hat{f}_\beta(M-2) \ + \ 2$$

Since $\hat{f}_\beta \leq |R|$, we get

$$\frac{\beta}{\alpha} \ \leq \ |R| \ (M-2) \ + \ 2,$$

which contradicts with our premise in (5), and thus proves that for $\hat{f}_\beta = \tilde{f}_\beta + 1$, satisfying the suggested condition in (5) guarantees that the prioritization condition in (2) holds, and thus concludes the proof. □

Appendix A.3. Proof of Proposition 3

There exist RWA-P instances for which the lower bound provided in Proposition 2 is necessary to prioritize request granting over link usage.

Proof. Consider an RWA-P network G being comprised of a pair of source and destination nodes s and t that are linked through two link-disjoint paths of length ℓ_a and ℓ_b. Suppose that there is only one request, i.e., $|R| = 1$, which is between nodes s and t, and the two distinct paths of length ℓ_a and ℓ_b together with some wavelength λ constitute the working and protection lightpaths for it, respectively.

Assume that $\alpha, \beta > 0$ are selected in such a way that for any pair of feasible solutions $(\hat{x}, \hat{y})$ and $(\tilde{x}, \tilde{y})$ with $\hat{f}_\beta > \tilde{f}_\beta$, the prioritization condition in (2) holds. Under this assumption, we want to show that the selected $\alpha, \beta > 0$ must satisfy the lower bound suggested in Proposition 2, which for the above-mentioned class of instances becomes

$$\frac{\beta}{\alpha} \; > \; |R| \; (M - 2) \; + \; 2 \; = \; \ell_a \; + \; \ell_b. \tag{A.9}$$

Let $(\hat{x}, \hat{y}) = (1, 1)$ and $(\tilde{x}, \tilde{y}) = (0, 0)$, giving $\hat{f}_\beta = 1$ and $\tilde{f}_\beta = 0$. Then, by (1), we have

$$\alpha \; (\ell_a + \ell_b) \; - \; \beta \cdot 1 \; < \; \alpha \cdot 0 \; - \; \beta \cdot 0,$$

giving the desired necessary condition of $\frac{\beta}{\alpha} \; > \; \ell_a + \ell_b$. □

Appendix A.4. Proof of Lemma 1

Dec-*RWA-P*-r *is in NP.*

Proof. Given an RWA-P-r instance $\mathcal{I} = (G, R, \{W^r\}_{r \in R}, \{P^r\}_{r \in R})$, where $G = (V, E)$ is a directed graph representing the optical network, R is the set of requests defined between pairs of source and destination nodes, and W^r and P^r are respectively the set of working and protection lightpaths for request $r \in R$, and a solution $\mathcal{S}$ to instance $\mathcal{I}$, we will show that we can verify in time polynomial in the size of $\mathcal{I}$ whether or not $\mathcal{S}$ properly grants k requests for some given integer $k \geq 0$. In what follows, we provide the complexity of each step in checking if solution $\mathcal{S}$ satisfies the constraints of the RWA-P-r problem and whether the number of granted requests is at least k.

1. For each request $r \in R$, we check if it is assigned (i) both a working and a protection lightpath or (ii) no lightpaths at all, which can be done by going over all working and protection lightpaths each time, which amounts to $\mathcal{O}\left(\sum_{r \in R} (|W^r| + |P^r|)\right)$ time in total. If the number

of granted requests is strictly less than k, the verification procedure terminates by concluding that solution $\mathcal{S}$ does not grant k requests. Otherwise, we continue verifying solution $\mathcal{S}$ with the following steps.

2. For each granted request, we check if the assigned working and protection lightpaths are link-disjoint by going over all tuples in conflict set $\mathcal{C}_1$. In the worst case, all possible pairs of working and protection lightpaths will be contained in $\mathcal{C}_1$ for each request $r \in R$, which makes the time complexity of this step $\mathcal{O}\left(\sum_{r\in R} |W^r| \cdot |P^r|\right)$.

3. For the lightpaths used in solution $\mathcal{S}$, we check if each pair having the same wavelength is link-disjoint. To this end, we go through the tuples in conflict sets $\mathcal{C}_2$, $\mathcal{C}_3$ and $\mathcal{C}_4$. In the worst case, all possible tuples will be contained in these sets, which yields the following complexities:

    - If the pair is comprised of a working and protection lightpath, we search through the tuples in $\mathcal{C}_2$, amounting to $\mathcal{O}\left(\sum_{r_1\in R}\sum_{r_2\in R}(|W^{r_1}| \cdot |P^{r_2}|\right)$ time in total.
    - If both lightpaths in the pair are working ones, we go through the tuples in $\mathcal{C}_3$, which takes $\mathcal{O}\left(\sum_{r_1\in R}\sum_{r_2\in R}(|W^{r_1}| \cdot |W^{r_2}|\right)$ time in total.
    - If both in the pair are protection lightpaths, we search through the tuples in $\mathcal{C}_4$, taking a total of $\mathcal{O}\left(\sum_{r_1\in R}\sum_{r_2\in R}(|P^{r_1}| \cdot |P^{r_2}|\right)$ time.

The time it takes to perform each one of the three steps listed above is polynomial in the size of the instance $\mathcal{I}$. Summing up the complexities of individual steps, we conclude that the overall complexity of verifying a given solution is polynomial in the size of $\mathcal{I}$ by the composition property of polynomials. Hence, Dec-RWA-P-r is in NP. □

Appendix A.5. Proof of Theorem 1

RWA-P-r is NP-hard.

Proof. We prove the hardness of RWA-P-r through a reduction from MSS. First, we describe a polynomial time reduction from a given MSS instance $G_s = (V_s, E_s)$ into an RWA-P-r instance $\mathcal{I} = (G, R, \{W^r\}_{r\in R}, \{P^r\}_{r\in R})$. Afterwards, we show that solving Dec-MSS in G_s is equivalent to solving Dec-RWA-P-r in $\mathcal{I}$; that is, finding a stable set of size at least k in G_s is equivalent to granting at least k requests in $\mathcal{I}$, for some given integer $1 \leq k \leq |V_s|$.

Construction. Given an MSS instance $G_s = (V_s, E_s)$, we begin with defining the request set R and an initial construction of G along with initial working and protection lightpath sets W^r and

P^r for each $r \in R$. Then, we form the conflict sets $\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3, \mathcal{C}_4$ using the edge set E_s of the MSS instance, and modify G and lightpath sets into their final form in accordance with the conflict sets. More specifically, we first create all link-disjoint lightpaths, and then modify them along with the network step by step, where in each step we create one distinct conflict from the conflict sets by modifying the involved lightpaths to have a common newly created link. Our reduction yields a simple directed graph G (i.e., we do not introduce multiple links between the same ordered pair of nodes although it is allowed), where there exist a unique working and protection lightpath for each request, the lightpaths pairwise have at most one link in common, and all of them have the same wavelength.

We divide our reduction procedure into four steps as explained in detail below.

1. *Requests.* We create $|V_s|$ requests, i.e., we set $R = \{1, \ldots, |V_s|\}$. Specifically, denoting the nodes in V_s by $n_r, r \in R$, for each MSS node n_r, we create a pair of source and destination nodes (s^r, t^r) for the optical network G and request r associated with it.

2. *Initial network and lightpath sets.* For each request $r \in R$, we create two nodes u_1^r and v_1^r, and generate two lightpaths between s^r and t^r, namely a working lightpath $w \in W^r$ with

   $$E[w] = \{(s^r, u_1^r), (u_1^r, t^r)\} \text{ and } \Lambda(w) = \lambda$$

   and a protection lightpath $p \in P^r$ with

   $$E[p] = \{(s^r, v_1^r), (v_1^r, t^r)\} \text{ and } \Lambda(p) = \lambda,$$

   where $E[\ell]$ and $\Lambda(\ell)$ respectively represent the set of links a given lightpath ℓ contains and the wavelength associated with it, as mentioned in Section 3.1.

   In words, the working and protection lightpaths for each request $r \in R$ and further all the lightpaths are initially link-disjoint, contain two links, and possess the same (arbitrary) wavelength λ (e.g., $\lambda = 1$).

   Figure A.4 illustrates an example MSS instance and the way a corresponding RWA-P-R network looks after performing steps 1 and 2 explained above, i.e., after constructing the request set R and the initial working and protection lightpath sets W^r and P^r for each $r \in R$. The nodes of the MSS instance G_s in part (i) of Figure A.4 are highlighted with different colors, and the same set of colors are used to indicate the source and destination nodes of the corresponding requests in the RWA-P-R network G in part (ii). The paths through nodes with u labels represent those of the working lightpaths, whereas the ones through v labels show those of the protection lightpaths.

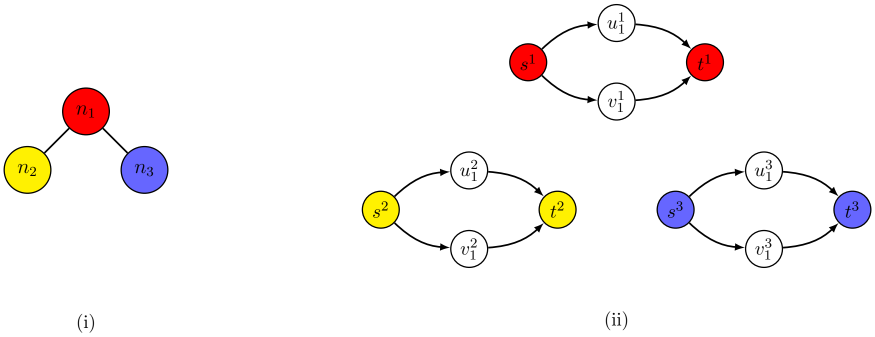

Figure A.4: (i) An MSS instance G_s, (ii) the initial incomplete form of the corresponding RWA-P-R network G.

3. *Conflict sets.* Next, we define the conflict sets $\mathcal{C}_1, \ldots, \mathcal{C}_4$, which we will utilize to modify the initial form of the network G. Specifically, for each pair of requests corresponding to an edge of G_s, we introduce different conflict tuples for $\mathcal{C}_i$'s, as detailed below:

(i) We define the working and protection lightpaths of any fixed request $r \in R$ to be link-disjoint. So, we have

$$\mathcal{C}_1 \;=\; \varnothing.$$

(ii) For each edge of G_s, we want the working and protection lightpaths of the associated requests to have a link in common. In particular, for each edge $\{n_{r_1}, n_{r_2}\} \in E_s$, we want the lightpath pairs (a) $w \in W^{r_1}$, $p \in P^{r_2}$, and (b) $w \in W^{r_2}$, $p \in P^{r_1}$ to share a link. As such, we define

$$\mathcal{C}_2 \;=\; \mathcal{C}_2^a \cup \mathcal{C}_2^b,$$

where

$$\mathcal{C}_2^a = \{(r_1, r_2, w, p)\colon r_1, r_2 \in R,\ \{n_{r_1}, n_{r_2}\} \in E_s,\ w \in W^{r_1},\ p \in P^{r_2}\}$$
$$\mathcal{C}_2^b = \{(r_2, r_1, w, p)\colon r_1, r_2 \in R,\ \{n_{r_1}, n_{r_2}\} \in E_s,\ w \in W^{r_2},\ p \in P^{r_1}\}.$$

(iii) For each edge $\{n_{r_1}, n_{r_2}\} \in E_s$, we want the pair of working lightpaths for the associated requests r_1 and r_2 have a link in common. Accordingly, we have

$$\mathcal{C}_3 \;=\; \{(r_1, r_2, w_1, w_2)\colon r_1, r_2 \in R,\ \{n_{r_1}, n_{r_2}\} \in E_s,\ w_1 \in W^{r_1},\ w_2 \in W^{r_2}\}.$$

(iv) Finally, for each edge $\{n_{r_1}, n_{r_2}\} \in E_s$, we want the pair of protection lightpaths for the associated requests r_1 and r_2 to share a link. So, we have

$$\mathcal{C}_4 \;=\; \{(r_1, r_2, p_1, p_2) \colon r_1, r_2 \in R,\ \{n_{r_1}, n_{r_2}\} \in E_s,\ p_1 \in P^{r_1},\ p_2 \in P^{r_2}\}\,.$$

We note that the total number of tuples in the conflict sets is $\mathcal{O}\left(|E_s|\right)$, because we generate a constant number of tuples for each edge in E_s.

In the example provided in Figure A.4, let $w^r \in W^r$ and $p^r \in P^r$ denote the working and protection lightpaths for request $r \in \{1,2,3\}$. Then, the conflict sets $\mathcal{C}_i$'s based on the MSS instance G_s in Figure A.4 are as follows:

$$\begin{aligned}
\mathcal{C}_1 &= \varnothing \\
\mathcal{C}_2 &= \left\{\left(1,2,w^1,p^2\right),\ \left(2,1,w^2,p^1\right),\ \left(2,3,w^2,p^3\right),\ \left(3,2,w^3,p^2\right)\right\} \\
\mathcal{C}_3 &= \left\{\left(1,2,w^1,w^2\right),\ \left(2,3,w^2,w^3\right)\right\} \\
\mathcal{C}_4 &= \left\{\left(1,2,p^1,p^2\right),\ \left(2,3,p^2,p^3\right)\right\}.
\end{aligned}$$

4. *Network and lightpath set modification.* The initial form of the RWA-P-R network G consists of the disjoint union of cycles (illustrated in part (ii) of Figure A.4), and hence does not yet incorporate the common links of lightpaths expressed in conflict sets $\mathcal{C}_1, \ldots, \mathcal{C}_4$. In this last step of the construction of the RWA-P-R instance $\mathcal{I}$, we modify G to make it consistent with the conflict sets specified above. To this end, we iterate over the tuples in the conflict sets, and modify and re-route the associated lightpaths so that they share a link. In alignment with the above-defined conflict sets, we keep the working and protection lightpaths of any fixed request link-disjoint, and modify each lightpath pairs appearing in the conflict sets such that they have one link in common. For ease of exposition, we use our running example in explaining the procedure. Let us consider the conflict tuple $\left(1,2,w^1,w^2\right)$, where $\left((s^1,u_1^1),(u_1^1,t^1)\right)$ and $\left((s^2,u_1^2),(u_1^2,t^2)\right)$ are the ordered set of links constituting the working lightpaths w^1 and w^2, respectively. We arbitrarily choose the lightpath w^1 to be the "host" to intersect w^1 and w^2 on, i.e., to create a shared link that currently belongs to the host. Starting with the reconstruction of the lightpath w^1, we delete the link (u_1^1,t^1), create a node labelled u_2^1, and add links (u_1^1,u_2^1) and (u_2^1,t^1), so that $\left((s^1,u_1^1),(u_1^1,u_2^1),(u_2^1,t^1)\right)$ becomes the revised working lightpath for request 1. Then, for lightpath w^2, we remove the link (u_1^2,t^2), create a node labelled u_2^2, and add links (u_1^2,u_1^1), (u_2^1,u_2^2) and (u_2^2,t^2), as a result of which the working

lightpath for request 2 becomes $\big((s^2, u_1^2), (u_1^2, u_1^1), (u_1^1, u_2^1), (u_2^2, t^2)\big)$, as highlighted in part (ii) of Figure A.5, with (u_1^1, u_2^1) now being the common link of the modified lightpaths w^1 and w^2. As an additional illustration of our lightpath reconstruction procedure, we next consider the conflict tuple $\left(1, 2, w^1, p^2\right)$ for our running example, arbitrarily select the host to be the lightpath w^1, and then perform a similar set of operations to re-route the lightpaths, the result of which is shown in part (iii) of Figure A.5. So, for each conflict tuple under consideration, we select one of the lightpaths to be the host, extend it by one link, and redirect the other lightpath through the newly created link. We note that this procedure establishes each lightpath intersection through newly created nodes and links in G, and thereby keeps the remaining lightpaths intact at every step.

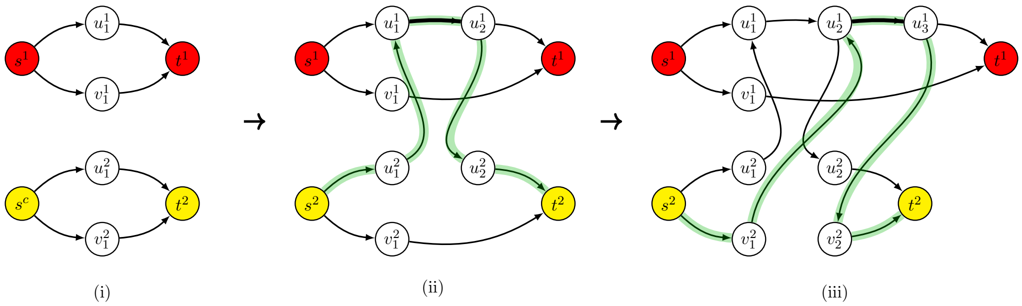

Figure A.5: Illustration of the re-routing of lightpaths in the RWA-P-R network G.

Reduction complexity. Let us investigate the complexity of building the RWA-P-R instance $\mathcal{I} = (G, R, \{W^r\}_{r\in R}, \{P^r\}_{r\in R})$ from a given MSS instance $G_s = (V_s, E_s)$. Steps 1 through 3 respectively take $O(|V_s|)$, $O(|V_s|)$, and $O(|E_s|)$ time. In step 4, we delete two links, add three links and two nodes for each conflict tuple, each of which takes constant time. Since there are $O(|E_s|)$-many tuples, we spend $O(|E_s|)$ time in total for step 4. Then, the complexity of instance reduction phase amounts to $O(|V_s| + |E_s|)$ in total, which is polynomial in the size of the MSS instance G_s. Hence, we have a polynomial time reduction from MSS to RWA-P-R.

Problem complexity. Now, we need to show that solving Dec-MSS in G_s is equivalent to solving Dec-RWA-P-R in $\mathcal{I}$. That is, we should show that there exists a stable set of size at least k in G_s if and only if we can satisfy at least k requests in $\mathcal{I}$.

($\Rightarrow$) Suppose that we are given a stable set of size k in G_s, say nodes $\{n_1, \ldots, n_k\}$ without loss of generality. Take lightpaths $\{w^1, p^1, \ldots, w^k, p^k\}$ in $\mathcal{I}$. If this selection of lightpaths satisfies the constraints of the RWA-P-R problem, then it means that we can grant k requests in $\mathcal{I}$. First, since there is only one working lightpath assigned to every request corresponding to nodes

$\{n_1, \ldots, n_k\}$, the constraint of assigning at most one working lightpath to each request is satisfied. Second, since no pair of nodes in $\{n_1, \ldots, n_k\}$ are linked by an edge, the corresponding lightpaths $\{w^1, p^1, \ldots, w^k, p^k\}$ do not share a link either, so the requirement that the working and protection lightpaths of each granted request being link-disjoint is satisfied as well. Finally, since we add both the working and protection lightpaths for each request r corresponding to the nodes of the given stable set, each granted request is properly assigned the two types of lightpaths. Therefore, we conclude that we can grant k requests in this case.

($\Leftarrow$) Conversely, suppose that we have a solution granting k requests in the Dec-RWA-P-r instance $\mathcal{I}$, say requests $\{1, \ldots, k\}$. Take the nodes in G_s that correspond to requests $\{1, \ldots, k\}$. Since all the lightpaths use the same wavelength in G, the ones associated with requests $\{1, \ldots, k\}$ cannot share a link. This implies that the nodes $\{n_1, \ldots, n_k\}$ representing requests $\{1, \ldots, k\}$ cannot be connected (i.e., they form a stable set), because if they were, the related lightpaths would have been forced to share a link by our construction.

This shows that solving Dec-MSS on G_s is equivalent to solving Dec-RWA-P-r in $\mathcal{I}$ built by a polynomial-time reduction from G_s. Since Dec-RWA-P-r is in NP, as shown in Lemma 1, we conclude that Dec-RWA-P-r is NP-complete, which implies that RWA-P-r is NP-hard. □

Appendix A.6. Proof of Proposition 4

When

$$\rho \; > \; \beta(|R| \; + \; 1) \; - \; \alpha \left(1 \; + \; \sum_{r \in R} \left(B^r_{w_{\min}} \; + \; B^r_{p_{\min}} \right) \right), \tag{9}$$

(8) *is an exact QUBO model for the RWA-P problem, where* $B^r_{w_{\min}} = \min_{w \in W^r} \{B^r_w\}$ *and* $B^r_{p_{\min}} = \min_{p \in P^r} \{B^r_p\}$.

Proof. Given the vector of working and protection lightpath binary decision variables x and y, respectively, we introduce g function to compactly rewrite the QUBO objective function expression in (8a) as

$$f(x, y) \; + \; \rho \; g(x, y).$$

That is, $f(x, y)$ is the objective expression of the IP as defined in (1a) (IP^{Base}), while $g(x, y)$ denotes the summation of the penalty terms in the QUBO objective before the common penalty coefficient of ρ is applied. Even though the feasible spaces of IP^{Base} and $\text{IP}^{\text{Strong}}$ are indeed the same, the IP model we actually refer to in this proof is IP^{Base} because our QUBO model is reformulation of that model. Note that $g(x,y) = 0$ if (x,y) is IP feasible, and $g(x,y) \geq 1$ otherwise by the construction of the penalty terms as explained before.

We first derive valid upper and lower bounds on the f function value over the set of IP feasible solutions, namely

$$F_{UB} \coloneqq \max\left\{f(x,y) \colon x, y \text{ binary and } g(x,y) = 0\right\}$$
$$F_{LB} \coloneqq \min\left\{f(x,y) \colon x, y \text{ binary and } g(x,y) = 0\right\}.$$

It is easy to observe that $F_{UB} = 0$ since (i) the trivial solution of accepting none of the requests yields $f(x = \mathbf{0}, y = \mathbf{0}) = 0$ for the maximization problem, thus $F_{UB} \geq 0$, and (ii) the selection of β and α values as suggested by Proposition 2 ensures that every granted request adds a negative term to the f value in the IP objective, hence yields $F_{UB} \leq 0$. On the other hand, the optimal value to the lower bound problem is at least the objective value that is obtained by granting all the requests using the shortest working and protection lightpaths for each request, yielding

$$F_{LB} \;\geq\; \alpha \sum_{r \in R} \left(\min_{w \in W^r} \{B_w^r\} \;+\; \min_{p \in P^r} \{B_p^r\} \right) \;-\; \beta \sum_{r \in R} 1 \;=\; \alpha \sum_{r \in R} \left(B^r_{w_{\min}} + \; B^r_{p_{\min}} \right) \;-\; \beta\;|R|.$$

This is again due to the way we set the values of α and β by Proposition 2.

Next, we observe that we can separate the range of the QUBO objective values of IP infeasible solutions from those of IP feasible solutions by pushing them beyond F_{UB}. As mentioned before, when a binary solution does not satisfy some IP constraints in (1b)–(1g), the magnitude of violation is at least one. Due to the way α and β values are set by Proposition 2, one additional unit of violation can lead to a decrease of the f value by at most $\beta - \alpha$, which happens when a zero value in the x vector is changed to one and y vector is kept intact, where the $-\alpha$ term is due to the resulting additional link usage being at least one. In order to guarantee that the QUBO objective values are shifted strictly above F_{UB} through penalty terms in case at least one of the IP constraints in (1b)–(1g) are violated, we set

$$\rho \;>\; \beta(|R| \;+\; 1) - \alpha \left(1 \;+\; \sum_{r \in R} \left(B^r_{w_{\min}} + \; B^r_{p_{\min}} \right) \right) \;\geq\; F_{UB} - F_{LB} + \beta - \alpha.$$

By selecting the penalty coefficient ρ such that it satisfies the above strict inequality, which is indeed the suggested one in (9), we obtain the following relations:

$$\begin{aligned}
(x, y) \text{ is feasible to the IP} \quad &\Longrightarrow \quad g(x, y) = 0 \\
&\Longrightarrow \quad f(x, y) + \rho \; g(x, y) \leq F_{UB} \\
(x, y) \text{ is infeasible to the IP} \quad &\Longrightarrow \; g(x, y) \geq 1 \\
&\Longrightarrow \quad f(x, y) + \rho \; g(x, y) > f(x, y) + F_{UB} - F_{LB} + \beta - \alpha \geq F_{UB}.
\end{aligned}$$

As such, the IP feasible solutions will always deliver smaller QUBO objective values than IP infeasible ones. Since our QUBO model is in minimization form, any optimal solution to the QUBO model must be one that is optimal (and feasible) for the IP model. □